Australian Government
Australian Cyber Security Centre

ACSC
AUSTRALIAN CYBER SECURITY CENTRE

iOS Hardening Configuration Guide

For iPod Touch, iPhone and iPad devices running iOS 9.3.5 or higher.

About this guide

This guide provides instructions and techniques for Australian government organisations to harden the security of iOS 9.3 devices.

Implementing the techniques and settings found in this document can affect system functionality, and may not be appropriate for every user or environment.

In these cases, organisations should seek approval for non-compliance from their accreditation authority to allow for the formal acceptance of the risks involved. Refer to *System Accreditation* and *Product Selection* chapters of the *Australian Government Information Security Manual* (ISM) for more information.

Evaluation status

Australian Signals Directorate (ASD) has completed an ASD Cryptographic Evaluation (ACE) of Apple iOS 9.3 scoped to review the data-at-rest and data-in-transit functionality and, when configured appropriately, has been found to be suitable for downgrading the handling requirements for data-at-rest and data-in-transit of PROTECTED information to that of UNCLASSIFIED.

Please refer to the consumer guide for further details on the scope and findings of the ACE. When completed, the consumer guide will be posted on ASD's website at:

http://www.asd.gov.au/infosec/epl/

This document is based on the findings of the ACE and provides policy advice that must be enforced for PROTECTED iOS device deployments. Guidance in this document will also assist agencies to comply with existing policies when deploying iOS devices at lower classifications.

Additionally, the latest version of Apple iOS on iPhone, iPad, iPad Pro and iPod Touch has completed evaluation of the Common Criteria *Mobile Device Fundamentals Protection Profile Version 2.0*, *Mobile Device VPN Client Protection Profile Version 2.0* and the *Mobile Device Management Client Protection Profile Version 2.0* at:

https://www.niap-ccevs.org/CCEVS_Products/in_eval.cfm

For more information on the evaluation process please refer to ASD's website:

http://www.asd.gov.au/publications/dsdbroadcast/20140410-evaluation-pathway-for-mobile-devices.htm

ASD expects a new major version of Apple iOS to be released in the near future. As usual, iOS 9.3 will no longer be available for download from Apple as a result.

When the new iOS version is released ASD will assess the security implications and provide additional guidance if required. In the interim, ASD advises the following:

- Upgrade to the latest version of iOS. Even though new versions were not the target of the evaluation, this version does provide security enhancements and addresses known vulnerabilities. This is consistent with ASD's advice to install the

latest versions of software and patch operating system vulnerabilities as communicated in the ISM and *Strategies to Mitigate Targeted Cyber Intrusions*.

- Implement the current *iOS Hardening Configuration Guide*. The existing guide is applicable to new versions of iOS and updated guidance will contain additions in response to new features, rather than wholesale changes to the existing advice.

As in any case where there have been significant updates of a previously released version of iOS, Apple provides detail of the content of security updates. This information may help agencies quantify the risk posed by not updating.

iOS and the Australian Government Information Security Manual

This guide reflects policy specified in the ISM. Currently, not all ISM requirements can be implemented on iOS 9.3 devices. In these cases, risk mitigation measures are provided in the *Risk Management Guide* at Chapter 11.

Chapter 6 provides recommended passcode settings for iOS devices. This advice has been developed based on an assessment of security risks related specifically to iOS 9.3 and A7 and later processors, and takes precedence over the non-platform specific advice in the ISM.

About the Australian Signals Directorate

As the Commonwealth authority on the security of information, the Australian Signals Directorate provides guidance and other assistance to Australian federal and state agencies on matters relating to the security and integrity of information.

For more information, go to http://www.asd.gov.au/about/

Audience

This guide is for users and administrators of iOS 9.3 or later devices. These devices include the iPod Touch, iPhone , iPad and iPad Pro.

To use this guide, readers should be:

- familiar with basic networking concepts
- an experienced systems administrator.

Parts of this guide refer to features that require the engagement of the technical resources of telecommunications carriers, firewall vendors or Mobile Device Management (MDM) vendors. While every effort has been made to ensure content involving these third party products is correct at the time of writing, agencies should always check with these vendors when planning an implementation.

Mention of third party products is not a specific endorsement of that vendor over another; they are mentioned as illustrative examples only.

Some instructions in this guide are complex, and if implemented incorrectly could reduce the security of the device, the network and the agency's security posture. These instructions should only be used by experienced administrators, and should be used in conjunction with thorough testing.

For further clarification or assistance, Australian government IT Security Advisors can consult ASD by emailing asd.assist@defence.gov.au or calling the ASD Cyber Hotline on 1300 CYBER1 (1300 292 371).

What's changed

iOS 9 has brought with it several important new features and improvements that are relevant to deployment in government and enterprises.

Device Enrolment Program

Although Device Enrolment Program (DEP) is not strictly an iOS 9 feature, as it became available in Australia during the time of iOS 8, it was not widely available from carriers and Apple Resellers supplying Government purchasing panels until late 2015. Frequently, but not always, devices purchased after 2 March 2011, but prior to availability of DEP can be retrospectively enrolled by the reseller. Whilst DEP is a free service from Apple, resellers can (but rarely do) charge a fee. A fee for DEP enrolment is most common when devices purchased earlier are retrospectively enrolled.

DEP is of significance for multiple reasons:

- it gives a level of supply chain assurance from factory to the end customer agency
- the setup assistant can be customised to only display selected screens, and users can enrol in MDM inside the Setup Assistant, without needing an App from the MDM vendor
- it can assign devices automatically to an MDM, without use of Apple Configurator 2

- it can ensure devices are placed in supervised mode, over the air, without use of Apple Configurator 2. Supervision enables additional policy controls, and management of capabilities such as Lost Mode and Activation Lock.

- it is the only way to make MDM enrolment mandatory and non-removable (even following a device wipe, the device is forced to re-enrol in MDM, whilst still inside the setup assistant).

Device Based App Assignment

Mobile Device Management Servers and Apple Configurator can assign institutionally licensed Apps purchased under the Volume Purchase Program directly to devices, without needing an AppleID on the device. An AppleID may be needed if other Apple services are used e.g. iMessage, personally installed Apps from the App Store or iCloud. Coupled with DEP and MDM, this significantly simplifies institutionally owned device deployment workflows.

Enterprise Developer Verification

The trust model of Enterprise in-House Apps has changed from iOS 8. Under iOS 9, Apps installed as the result of an MDM command lead to the associated Enterprise Developer code signing certificate being implicitly trusted. If users manually install Enterprise signed Apps, they now need to navigate to the Settings App, and explicitly trust the signing certificate. There is no longer a dialog at App launch that lets them trust the certificate. Devices also periodically check a list of revoked Enterprise Developer code signing certificates available at ppq.apple.com. These changes are designed to make it more difficult for users to be socially engineered to install Enterprise Apps from non-trusted sources. This means that even with deployments relying solely on Enterprise Apps, devices MUST be able to reach ppq.apple.com, or devices need to be periodically re-imaged from Apple Configurator 2 (as the Enterprise Apps will eventually fail to launch if they repeatedly can't reach ppq.apple.com).

App Transport Security

In iOS 7 and iOS 8, Apple transitioned the default data protection class of all Apps to "ProtectionCompleteUntilFirstUserAuthentication", which is semantically similar to full disk encryption. This assists in mitigating the security risk of a jailbreak being used to access App data. Developers can opt in to more restrictive data protection classes e.g. ProtectionComplete, which is only accessible when the device is currently unlocked. This data protection class MUST be used for data classified PROTECTED, and is available as a toggle in Xcode for the developer at compile time. If an App is required to write PROTECTED data while in a locked state, such files must be protected using Class B NSFileProtectionCompleteUnlessOpen.

In iOS 9 and Xcode 7, Apple has attempted to address the data-in-transit issue, and introduced App Transport Security (ATS). ATS forces the default transport security for any App using the SecureTransport APIs (typically NSURLSession), to use TLS 1.2 with forward secrecy ciphers. These are *ASD Approved Cryptographic Protocols* (AACP) which use *ASD Approved Cryptographic Algorithms* (AACA). Until 31 December 2016, developers had the ability to explicitly downgrade the

transport security of an App. This is documented explicitly in the App's "exception.plist". App developers MUST NOT disable ATS for PROTECTED data. After 01 January 2017, all Apps submitted to the App store must use ATS. If a developer does not use SecureTransport or ATS, the App's method for securing data-in-transit must comply with the relevant ISM controls.

New Configuration Profile Controls

New management and supervisory controls have been made available to iOS enterprise fleet administrators. Refer to *Recommended Device Profile Settings* for our updated advice.

Improved VPN functionality

iOS 9 contains several under-the-hood changes to VPN behaviour with the new Network Extension framework. The most significant change for this guide, is that the built in IPSec IKEv2 VPN agent is now available for use in a per-App VPN configuration, in addition to Always On VPN configuration. The IPSec IKEv2 VPN client is evaluated and is the preferred VPN transport. Refer to the *VPN* section for detail.

Exchange ActiveSync version 16

iOS 9 supports Exchange ActiveSync version 16 (EAS 16), which includes a significant re-write of the calendaring protocol. EAS 16 is currently available though Microsoft's Office 365 service, and is expected to become available for on-premises deployments in an update to Windows Server some time in 2016. On iOS, EAS 16 brings Exchange calendaring up to feature parity with CalDAV, and for the first time allows for attachments to calendars to be synced to a mobile device. This is significant, because whilst the native Mail App in iOS is suitable for holding attachments with PROTECTED content, at the time of writing, the native Calendar App is only suitable up to FOUO or UNCLASSIFIED with DLM. Agencies should consider the residual security risk and mitigation measures when upgrading servers to an Exchange version that supports EAS 16.

Lost Mode

iOS 9.3 enables a supervised device to be placed in "Lost Mode". This mode turns on Location Services (even if it has been disabled), and reports the device location to the MDM server. The device lock screen displays that the device is in Lost Mode. MDM can also disable Lost Mode. Once the device is unlocked by the user, they are presented with a modal dialog that states that the device was placed in Lost Mode at a specific date and time. Location Services is returned to its previous state when the device exits from Lost Mode. No Apple ID is required for this functionality, just supervision and MDM (noting that it is similar to Lost Mode controlled at a user level provided by the Find my iPhone capability in iCloud).

Feedback

Advice has been updated throughout the guide based upon the experiences of agencies and industry.

If you have feedback email: asd.assist@defence.gov.au

iOS Hardening Configuration Guide

CHAPTER 1

Introduction to Mobile Device Security

Understand the key technologies that can be used to secure iOS mobile devices.

Introduction to Mobile Device Security

This chapter provides the planning steps and architecture considerations necessary to set up a secure environment for mobile devices. Much of the content in this chapter is platform agnostic, but some detail is written to address specific features available in iOS. Not all of the options discussed will be applicable to all environments. Agencies need to take into account their own environment and consider their acceptable level of residual security risk.

Assumptions

This chapter makes some basic assumptions regarding the pervasive threat environment:

- at some point, there will be no network connection present
- all radiated communication from the device has the potential to be monitored
- all conventional location, voice and SMS/MMS communications are on an insecure channel
- certain infrastructure supporting mobile devices can be trusted
- carrier infrastructure cannot always be trusted as secure in all countries
- at some point, devices will be lost or stolen
- lost or stolen devices will be in a locked state
- third party Apps will leak sensitive or classified data.

Offline device security

When a device is offline, protection of data on the device is determined by how the device implements policy locally. There can be no referral to a server for policy or any remote wipe command if there is no network present.

When offline, the security of the device is determined by:

- policy enforced by iOS from Exchange ActiveSync (EAS) or Configuration Profiles
- policy enforced by third party mobility management Apps
- the security settings set locally on the device (such as passcode policy and USB pairing restriction)
- the device's cryptographic capabilities
- the correct use of file protection classes and Keychain by Apps
- the strength of the device passcode.

Device security on the network

The general principle that applies for all data when the device is on a network is that wherever possible, all network traffic should be encrypted, noting that all classified network traffic passing over a connection that is not afforded sufficient physical

protection (i.e. public network or connections between sites) must be encrypted as per the *Cryptography* chapter of the ISM. This is not usually achieved merely by turning on a Virtual Private Network (VPN) for all traffic. Typically this involves using a mixture of:

- TLS to encrypt connections to specific hosts such as mail servers or management servers that need to be highly reachable
- TLS for any traffic containing sensitive or classified data
- a VPN for more general intranet access
- WPA2 with EAP-TLS as a minimum for Wi-Fi security
- 802.1X authentication on Wi-Fi networks combined with Network Access Controls to compartmentalise Wi-Fi access to defined security domains.

Some third party mobility management vendors may provide an operating system integrity check via an App. Though a useful feature from a compliance perspective, it cannot be relied upon, and can lead to a false sense of security. The App doing the integrity check is running as an unprivileged user, and therefore is readily deceived if there has been a kernel level compromise of the device, commonly known as a jailbreak. Jailbreaking is better defeated by supervision, blocking host pairing and passcode policy. Details and mitigations against jailbreaking are available in the *Jailbroken Employee Owned Devices* section.

Apple Push Notification Service

Many Apps and services associated with iOS devices take advantage of the Apple Push Notification Service (APNS), in order to minimise background activity and extend battery life. APNS acts as a trusted third party, sending a “ping” to a specific App, letting it know to “phone home” to its servers. In addition, APNS can be sent small notifications, such as updating the badge on an icon, playing an alert tone or displaying a short text message.

Apple’s documentation on APNS refers to servers that communicate with Apps as “Providers”.

Example of providers include push email notifications, Mobile Device Management (MDM) servers, Enterprise Middleware for SAP, IBM and Oracle, and iOS client-server applications that are able to execute in the background (such as VoIP Apps, streaming audio Apps or location tracking Apps). Providers send a request to the device to “phone home”, and the App or agent on the device establishes communication with and responds to the Provider. For example, MDM servers send a “phone home” notification via APNS to the Apple MDM agent on the device. The Apple MDM agent then “phones home”, establishing a TLS tunnel directly to the MDM server, and exchanging XML queries and responses inside this tunnel.

It will be necessary to set appropriate firewall rules to enable APNS. Refer to *Firewall Rules* in Chapter 12 for information on ports and services.

Data roaming

Data roaming generally refers to a process by which your cellular device is able to receive data on mobile networks that your telecommunications operator doesn't own.

There are two main security risks associated with data roaming:

- When roaming internationally, there are both implied and actual lower levels of trust of the foreign network. As soon as traffic goes international, it is no longer subject to the privacy and consumer protection requirements that apply to purely domestic communications in the host country. It is incorrect to assume that the rights protecting an individual's privacy are uniform internationally. For more information see ASD advice on travelling overseas with an electronic device at http://www.asd.gov.au/publications/protect/electronic_devices_os_travel.htm
- If data roaming is switched off for cost management, then the device is "off the grid" for management and monitoring consoles such as EAS, MDM or iCloud's "Find My iPhone", unless connected to a Wi-Fi network.

iOS devices may be configured to disable voice and data roaming via MDM, but it is possible for users to re-enable the global roaming settings on their devices. Note that:

- Users can fine tune which Apps use cellular data from Settings, as well as get detailed visibility on what Apps and services are using cellular data. MDM can view, but not change these settings.
- MDM can prevent any or all Managed Apps from using cellular data, without disabling data roaming globally. This means roaming data usage is primarily caused by user action.

If an agency uses a custom APN domestically (e.g. for billing or traffic prioritisation purposes), when devices roam internationally the carrier they roam to may or may not support extending the domestic carrier's APN to their network, and devices may lose connectivity. Devices which are likely to be used broadly internationally, or are known to be used in countries where APNs do not roam to, generally should not be configured with a custom APN (Always On VPN can establish a secure tunnel without use of a custom APN).

Apps

As outlined in ASD's *Strategies to Mitigate Targeted Cyber Intrusions*, ASD recommends that enterprises actively control which software applications are permitted to run on a computing device. Note that iOS 9.3 supports explicit whitelisting or blacklisting of Apps by App Identifier. There are a number of ways for institutions to procure and load Apps onto an iOS device.

- Institutions can purchase Apps in Volume under Apple's Volume Purchase Program (VPP), and deploy to devices via MDM or Apple Configurator 2. Generally this should be done via Device Based App Assignment, which has no association to an Apple ID.
- They can also purchase customised version of App Store Apps under the B2B program, and deploy to devices via MDM or Apple Configurator 2.

- They can deploy Apps signed by their own Enterprise Developer code signing certificate, and deploy to device via MDM or Apple Configurator 2.

Deploying Apps on mixed use devices must be done via MDM, as this deploys Apps as Managed, and permits Managed Open-In controls to be applied, and Apps to be configured over the air.

User self install

If a device does not have host pairing blocked, a user can compile and self install Apps using Xcode, over a USB tether, provided both the iOS device and Xcode are using the same Apple ID.

Generally, agency owned devices should be supervised, and host pairing should either be blocked, or constrained to a specific whitelist of trusted identities. These settings prevent a user from installing Apps via this method.

If an Apple ID is permitted on the device, and access to the App Store is permitted, then users have the ability to install Apps from the iOS App Store.

App Store

The App Store is hosted and curated by Apple, and is focused on mass-market distribution of paid and free Apps. Apps are potentially loaded to a device:

- Over-the-air (OTA) from the App Store itself (there is a 100 MB size limit if cellular data is used, 4 GB over Wi-Fi)
- OTA via a Mobile Device Management (MDM) solution
- via USB on a paired host running Apple Configurator 2
- via USB/network on a paired host computer running iTunes.

Apple maintains discretionary control of curating App Store content, and can remove Apps for a variety of reasons. ASD recommends that corporately provisioned Apps are tested and approved by the agency prior to use.

Although App Store Apps come from a curated environment, and the runtime environment the Apps execute in is a relatively hardened one, Apple's review process is focused on end user privacy, and does not implement ISM policy. Agencies should assess the security risks associated with allowing unrestricted user-initiated installation of Apps, building on Apple's review process. Some additional security risks that need to be considered are:

- the inappropriate use of data protection
    - Most App Store Apps default to the Class C data protection class, PROTECTED data must be stored in Class A or Class B.
- the inappropriate use of transport security
    - Ensure that Apps use iOS native TLS APIs and mitigate against man-in-the-middle attacks using constrained trust chains or certificate pinning where possible.

    - Use of NSURLSession APIs ensures a default of TLS 1.2 with forward secrecy ciphers, and it is simple to extend this to constrained trust chains where only specific CAs are trusted.

- the inappropriate requests for access of private information, contact list, photos or location information

    - Note that these can be allowed/denied dynamically by the user.

- synchronisation and/or storage of data with cloud services

    - Note that while Apple supports blocking iCloud services selectively from MDM policy, use of services like DropBox, Skydrive and Google Drive need to be managed either on a per-App basis or blocked at the network layer.

- the inappropriate registration of Uniform Type Identifiers (UTIs) and URL handlers

    - Note that iOS 9 limits the number of UTIs that an App can register for.

Agencies can manage these security risks through discussions with the App developer or through conducting professional penetration testing. Refer to *Managing Apps and Data* in Chapter 5 for information on questions to ask App developers.

Note: App Store Apps usually are automatically updated over the air using Wi-Fi, but may update over cellular if under 100 MB and device settings allow this.

In-House Apps

Through the use of an ad-hoc provisioning profile, up to 100 instances of a signed App binary can be installed.

Ad-hoc Apps are locked to a specific set of devices by the provisioning profile. These are most commonly used for beta testing of Apps, or where very restricted distribution of a small number of instances of a bespoke App is appropriate.

Agencies with a Dun and Bradstreet Data Universal Numbering System (DUNS) number can apply to become Enterprise Developers. This allows the creation and distribution of custom Apps and provisioning profiles within an agency for its own use, where the distribution is limited to employees and contractors (i.e. not to the general public). There is no limit to the number of instances of such Apps.

Enterprise In-House Apps can be installed using:

- Apple Configurator 2 (USB only)
- OTA via website
- OTA via MDM server
- iTunes (USB and Wi-Fi).

In all the above cases, an MDM console allows monitoring of versions of Apps installed on a device. This allows a management decision as to when updates are required. Enterprise Apps can also be built to self update.

Note that as anyone with a valid Enterprise Developer code signing certificate can sign Apps, and host them on a web site,

there is potential for users to be socially engineered to install such Apps.

iOS 9 contains several mitigations for attacks attempting to take advantage of this:

- Apps can not be installed silently over USB when tethered.
- MDM can toggle a restriction preventing the user from trusting any Enterprise Developer code signing certificates MDM did not install. Any Apps installed by MDM are implicitly trusted. ASD strongly recommends this setting be applied.
- MDM can toggle a restriction allowing or preventing the user from installing any Apps at all (with all Apps being installed as a result of MDM commands). Agencies may enable this subject to a risk assessment and appropriate mitigations being in place.
- If the user is allowed to self install Enterprise Apps, then unlike earlier versions of iOS (where the user could trust the App on launch), they need to go into Settings, and explicitly trust the Enterprise Developer code signing certificate associated with that App. Generally, agencies should not allow this but there can be exceptions where the security risks are acceptable.

Volume Purchase Program (VPP)

The VPP allows businesses to buy App Store Apps in bulk using a corporate purchasing card. Agencies with a VPP account may also use the B2B portal to request custom versions of App Store Apps directly from App developers. Agencies may request custom Apps that utilise a stronger use of data protection or transport security APIs, or request that functionality (such as cloud synchronisation) be disabled.

Historically, Apps purchased through VPP were redeemed by a user to a particular Apple ID permanently. Recent changes to VPP now allow Apps to be both assigned to and revoked from users, or from devices directly, without association to an Apple ID.

For more information on VPP go to http://www.apple.com/au/business/vpp/

Managed Apps

App Store and Enterprise In-House App installations can be triggered via an MDM server; these Apps are called “Managed Apps”. Managed Apps can be uninstalled by the MDM server along with any associated data or can be set to uninstall when the MDM profile is removed. Apps containing PROTECTED data should be managed.

Web Apps

The iOS web browser Mobile Safari has extensive support for HTML5, CSS3 and JavaScript features for Web Apps. This is often a useful mechanism to deploy informational Apps quickly

from a central intranet point; however Mobile Safari on iOS is still subject to the same threats as other browsers, and only stores its caches in “AfterFirstUnlock”. Therefore, hybrid Apps that use an MKWebView to lock to a web site, may require security features appropriate for PROTECTED data to be applied.

iTunes

iTunes is no longer a requirement for device management although it can be useful in some deployment models, particularly as a mechanism to backup devices. If agencies decide to use iTunes as part of their device management workflow it can be locked down for use on agency Standard Operating Environments (SOE) via registry keys or XML property lists as detailed at:

http://help.apple.com/iosdeployment-itunes/mac/1.2/

Apple IDs

There are two common misconceptions around Apple IDs:

- they are required to install Apps.
- they must be associated with a credit card.

For a Bring Your Own Device (BYOD) model, there is generally implied trust that users can continue to install Apps on their own device. Therefore, users may continue to use their existing Apple ID. It may be appropriate to register this Apple ID as part of the process of submitting to the agency acceptable use policy. An MDM console can be used to monitor what Apps have been installed. MDM can push Apps to devices without associating the App licence with the Apple ID on the device (called “device based App assignment”), or it can associate the App with the Apple ID (in which case any device with that Apple ID active could download the App from the App Store). Generally, device based assignment is preferred.

For an agency owned device model, Apple IDs do not need to be created to install Apps.

Where agencies do want to allow use of Apple IDs, for services such as iMessage, Find my iPhone or Activation Lock, a decision needs to be made if users use their personal Apple ID, or an agency-specific one. For limited personal use policy, where the user may be allowed to install Apps as unmanaged, then the user’s existing Apple ID is usually most appropriate. If agency-specific Apple IDs are created, they are best done on device, using a process that ensures they are not linked to a credit card. The process for doing this is described at:

http://support.apple.com/kb/HT2534

If an agency plans to create a large number of AppleIDs in a short period of time, such as during a deployment, they should contact Apple’s Enterprise and Government sales team to ensure whitelisting of the agency from the real time anti-fraud blacklisting provisions of the Apple Store infrastructure.

If device based App assignment is used, an agency can set policy to prevent the user from creating an Apple ID. In this scenario, iCloud and iTunes services cannot be used, and the only Apps installed are provisioned by MDM or Apple Configurator 2.

Siri

Siri provides voice to text services by transmitting voice data to remote services for processing. Only UNCLASSIFIED dictations may be performed using Siri. The identifier used to tag Siri data on Apple's servers is randomly generated by the device, and can be changed, but data is retained for 6 months. By default Siri can be used from a locked screen to perform actions such as opening emails and reading calendar entries. This behaviour can be disabled via Configuration Profile restriction while still allowing Siri when unlocked.

Lock Screen Services

By default locked iOS devices still provide some limited services for user convenience such as:

- Control Centre
- Notification Centre
- Today View
- Siri.

These lock screen services may be administratively disabled, although in doing so there is a security versus usability trade off, e.g. it may be possible to allow Today View, with appropriate detailed configuration so only UNCLASSIFIED information is displayed.

Control Centre

The Control Centre panel is accessed at the lock screen with an upward swipe from the bottom of the screen, and contains the following:

- airplane mode, Wi-Fi, Bluetooth, Do-Not-Disturb mode, rotation lock and flashlight on/off switches
- screen brightness
- volume and media playback controls
- AirDrop discoverability control and AirPlay output selection
- access to Stopwatch, Calculator and Camera Apps.

Although the new Control Centre has many useful user convenience functions, there are security risks in enabling it from the lock screen, e.g. AirDrop and Airplay controls may provide data leakage paths (as both can run over ad-hoc networks). Control Centre can be disabled at the lock screen via Configuration Profile restriction, but it will still be accessible after the device has been unlocked, where the decision to use them can be managed by an authorised user.

Notification Centre

Notification Centre and Today View are accessed from the lock screen with a downward swipe from the top of the screen. iOS displays APNS notifications at the lock screen such as messages (iMessage) or notifications from Apps. Some Apps support interactive notifications, where the user can provide a response, or a reply, without unlocking the device.

Although such content must be UNCLASSIFIED, revealing notifications at the lock screen is not recommended. Notification Centre from lock screen can be disabled via Configuration Profile restriction.

Today View

Today View gives a summary of information about a user's day. Normally this will include the day's weather, traffic and calendar entry.

Today View can be disabled via Configuration Profile restriction.

Planning your mobility deployment

Every agency should have a BYOD policy, even if that policy is "no employee owned devices are to be allowed on the agency network". ASD's *Risk Management of Enterprise Mobility including Bring Your Own Device (BYOD)* provides advice regarding the security risks and business benefits of a number of possible mobility approaches. It comprehensively covers:

- the potential business benefits of enterprise mobility
- how to choose an appropriate mobility approach, including:
    - non-personalised devices, that are potentially shared use, but contain no information personalised to a user identity
    - corporately owned devices "Corporately Owned, Personally Enabled" (COPE - allowing limited personal use)
    - corporate employee managed "Choose Your Own Device" (CYOD - usually a limited selection)
    - employee owned devices "Bring Your Own Devices" (BYOD - usually restricted to a limited selection)
    - developing mobility policy
    - risk management controls.

This document is a thorough guide to planning a mobile deployment and is available at:

http://www.asd.gov.au/publications/csocprotect/enterprise_mobility_bring_your_own_device_byod_paper.htm

CHAPTER 2

Security Features and Capabilities

Mobile device security features, and the enabling technologies for implementing those features under iOS.

Security Features and Capabilities

This chapter covers mobile device security features, and the enabling technologies for implementing those features under iOS and related infrastructure.

Security features in iOS

iOS provides a number of security features that include:

- management of credentials and passwords with Keychain
- encryption of data-at-rest using data protection classes (which uses AACA and AACP)
- encryption of data-in-transit using Secure Transport (which uses AACA and AACP)
- encryption of data-in-transit using IPSec IKEv2 VPN (which is evaluated and uses AACA and AACP)
- digital signatures, certificates and trust services
- randomisation services
- code signed applications
- sandboxing.

B2B or Enterprise In-House Apps developed for an agency should generally take advantage of these services, rather than re-inventing the same capabilities.

Configuration Profiles

Configuration Profiles are XML formatted plist files that contain device settings, security policies and restrictions. An administrator may use a Configuration Profile to:

- set passcode policy on a device
- set restrictions (such as disabling use of Siri)
- configure wireless networks
- configure VPN
- configure email
- install X.509 certificates
- set an MDM) server.

These are only a few examples of possible configuration options. For more information please see the iOS Configuration Profile Reference:

https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/

Note: Configuration Profiles are not encrypted by default. MDM Servers copy Configuration Profiles on a device through a TLS 1.2 tunnel.

Credentials that are stored in Configuration Profiles are available to anyone who has access to the files. Configuration Profiles that are manually installed support two behaviours:

- If the profile is not encrypted, password fields are ignored at manual install time, and the user is prompted to enter the password on device when the profile is installed.
- If the profile is encrypted (or sent via MDM) it can contain passwords at manual install time on the iOS device. Passwords may be in cleartext or encoded in Base64, which offers no protection (beyond any transport layer protection used). Use of identity certificates is often preferable to username/password combinations, as on device, they are tied to the device's hardware key.

Some of the credentials that could be in a Configuration Profile include:

- Wi-Fi passwords
- VPN shared secrets
- Email usernames/passwords
- ActiveSync usernames/passwords
- LDAP usernames/passwords
- CalDAV usernames/passwords
- CardDav usernames/password
- Subscribed Calendars usernames/passwords
- SCEP pre-shared secrets.

Note: Take care to ensure that Configuration Profile files are stored appropriately and not improperly accessed. Many MDMs manage profile creation dynamically, move them to the device using TLS and support use of identity certificates in preference to usernames and passwords.

Enterprise distribution

Provisioning profiles allow custom applications to be run on iOS devices. They are often used in the following ways:

- to allow developers to test Apps on their devices
- to allow agencies to distribute Apps directly to their employees and contractors.

Note that Enterprise Apps need to periodically check a certificate revocation list hosted at ppq.apple.com, regardless of how they are deployed, and will fail closed if the check fails, rendering the App inoperable. Note that Enterprise Apps installed by MDM are implicitly trusted, whereas Enterprise Apps installed by users require explicit trust to be enumerated (and the user ability to approve this trust can be controlled by a restriction payload in a Configuration Profile). For more information please see:

https://support.apple.com/en-au/HT204460

To obtain an enterprise distribution code signing certificate, an agency must join the Apple Developer Enterprise Program.

More information about the Developer Enterprise Program can be found at:

http://developer.apple.com/programs/ios/enterprise/

Note: If the enterprise program login is compromised an adversary could install malicious applications on users' iOS devices. Use of two-factor authentication on this account is strongly recommended.

Find My iPhone and Activation Lock

Find my iPhone (FmiP) is an iCloud service which allows users to locate and remote wipe their devices. On devices which have not been configured as supervised, FmiP enables a feature called "Activation Lock". This feature ties the Apple ID used for FmiP to device activation. Following a device wipe or OS installation, a user must enter their Apple ID credentials. This is intended to reduce the resale value of stolen devices, as the device cannot be used without knowledge of the original user's Apple ID.

Activation Lock has the potential to deny an agency access to its own devices, if they are not deployed in supervised mode. If Activation Lock is enabled on an agency owned iOS device that is not supervised, with a user's personal Apple ID the user must disable FmiP prior to returning the device for service. If the device were returned with Activation Lock enabled, the agency would not be able to re-activate after re-installing iOS, and may not remove FmiP even if the device passcode was known.

By default, supervised devices have Activation Lock disabled however, agencies may choose to re-enable this through the use of MDM. Use of supervision also allows the MDM server to hold an Activation Lock bypass key in escrow, and unlock the device over the air on command. Please refer to the *Activation Lock* section of the *Deploying iOS devices* chapter for further information.

Use of DEP, and requiring authenticated user enrolment in MDM in the setup assistant is another way to deny access to use of a device to unauthorised users, even if it has been wiped. It can be used in conjunction with Activation Lock.

The link below contains information on how personal users may take advantage of the service:

http://www.apple.com/au/icloud/find-my-iphone.html

Sandboxing

Sandboxing ensures that Apps that run on iOS devices are restricted in terms of the resources that they can access. This is enforced by the kernel.

For detailed information on the iOS/OS X sandbox see Jonathan Levin's presentation "Hack in the (sand)box" http://newosxbook.com/files/HITSB.pdf

When devices are enrolled in to MDM, Apps and App Extensions obey "Managed Open-In" restrictions. This feature allows administrators to configure devices in a way that prevents documents being exported from "managed" to "unmanaged" (user) Apps. It can also be used to prevent

Unmanaged App Extensions being accessible from Managed Apps. These native features have reduced the need for the use of managed container Apps; however, there may be cases where an agency may seek a third party solution.

In cases where the agency requires greater protection of their contact and calendar information a third party solution may be required. These would need to be risk assessed by the agency in accordance with the ISM, as they may not use Data Protection, the built-in iOS cryptographic libraries, or indeed even use AACP or AACA.

A number of managed container solutions can be used to provide:

- finer grained control and policy enforcement for data transmission between Apps which support the third party SDK
- a suite of Apps which have been designed to enforce a specific secure workflow

There are a number of non security related trade-offs which should also be considered before using third party container solutions:

- third party container solutions often have a lower level of integration with both Apps present in iOS and common App Store Apps
- many third party container solutions rely upon software encryption which is both slower, and degrades battery life, compared to those which rely upon the hardware accelerated native data protection offered by iOS
- third party container solutions which rely upon software encryption may require an additional passcode separate to the device passcode, often with an increased complexity requirement
- Some third party container solutions have been observed to use Class D Data Protection (aka ProtectionNone), thereby purely relying on software encryption. On devices without host pairing blocked, these data files are susceptible to offline attack, and are potentially in a weaker data-at-rest posture than files outside the container in an appropriate data protection class
- Some third party container solutions and wrapping SDKs have a track record of being fragile when iOS updates are applied. The loss of functionality when the current version of a container is incompatible with the current version of iOS can deter agencies from updating to the current iOS version, which violates one of the mandatory Top 4 mitigations from ASD's *Strategies to Mitigate Targeted Cyber Intrusions*.

When used, care must be taken to prevent a situation where the agency provided container solution is so prohibitive that users attempt to start working their way around the system.

Information regarding evaluated container solutions is available on the EPL at:

http://www.asd.gov.au/infosec/epl/

Content filtering

Access to intranet sites and some mail, contact or calendar data can be achieved via reverse proxies and content filters. There are multiple solutions in this space.

EAS filtering products can be used to ensure email sent to EAS devices have appropriate protective markings for the classification the device is approved to by an agency. This approach allows mobile devices to only receive email content at a classification appropriate to the device, as well as having policy and controls applied to the email content.

In this scenario, the agency's Wide Area Network (WAN) security domain is not extended out to the mobile device, and there is no need to lower the classification of the agency WAN. Such solutions can be used to redact specific content patterns from emails sent via EAS, for example, to scrub credit card numbers from all emails synced to mobile devices. This class of tools can also facilitate correct protective marking of email coming from mobile devices without direct on-device support for Australian government protective marking standards. For further information see the ISM section on *Content Filtering*.

Enterprise Single Sign On (SSO)

Enterprise SSO helps to reduce the number of times a user is required to enter credentials, particularly when switching between Apps. It also can support moving credentials from a relatively simple to attack form (username and password) to a more cryptographically robust form (identity certificate). SSO is built on top of Kerberos, which is the industry standard authentication scheme already present in many corporate networks.

To take advantage of SSO, agencies must create and then deploy an Enterprise SSO Configuration Profile payload. This payload contains:

- username (PrincipalName)
- password (optional)
- identity certificate (optional)
- Kerberos realm name
- a list of URL prefixes where Kerberos authentication should be applied
- a list of App identifiers which will be granted access to Kerberos.

Some Apps support SSO without modification, others may have to be updated to take advantage of Apple's higher level networking APIs (primarily the NSURLSession family).

In iOS 9, if Kerberos is used in conjunction with per-App VPN, the Kerberos transaction occurs inside the VPN tunnel, and a Kerberos Key Distribution Centre Proxy (KKDCP) is not required - the KDC merely needs to be accessible from the VPN endpoint.

For additional details please refer to the Apple WWDC 2013 presentation: *Extending Your Apps for Enterprise and Education Use* (Apple Developer login required):

http://devstreaming.apple.com/videos/wwdc/2013/301xcx2xzxf8qjdcu3y2k1itm/301/301.pdf

Per-App VPN (PAVPN)

In deployments which permit limited personal use, per-App VPN can be used to divide work and personal network traffic. Typically, this would be configured so that corporate network traffic from Managed Apps traverses a VPN back to the corporate intranet, while unmanaged personal Apps connect to the internet directly or through a proxy. This VPN enhancement is intended to increase the privacy of the user by not transmitting their personal network traffic through corporate infrastructure while at the same time protecting the corporate network from traffic generated by Unmanaged Apps.

Per-App VPN can be combined with Kerberos SSO in many situations.

Per-App VPN can be configured for either the built in IPSec IKEv2 VPN client, or via third party VPN client Apps (available from both VPN vendors and some MDM vendors). IPSec IKEv2 is currently the only evaluated VPN option.

Always On VPN (AOVPN)

When the iOS device is primarily containing PROTECTED data, with very limited personal use, or it frequently needs to connect to untrusted networks, Always On IPSec IKEv2 is the preferred VPN configuration. Always On VPN is the highest level of protection to network traffic possible from native iOS tools, and is part of the Common Criteria MDFPP and AISEP Target of Evaluation (TOE). Note that this VPN configuration routes all traffic, and exceptions need to be whitelisted to allow a device to talk to telephone carrier voicemail servers, MMS and similar services that are normally outside a VPN tunnel. This VPN configuration fails closed, so if the tunnel can not be established, the device has no network connectivity. Because all traffic from the device goes down the tunnel, the VPN end point needs to be designed as an enclave network that allows traffic to Apple for APNS, certificate validation and similar security and trust services integral to the operation of the device.

Use of AOVPN also affords the following management advantages:

- Proxy configuration can be greatly simplified by using a transparent proxy.
- Explicit control of device updates can be controlled by blocking access to mesu.apple.com at the network layer. When an update is approved, this block is removed, and MDM sends an “updatenow” command to the device.

Design of the enclave network is critical to effective functioning of the devices, and needs to be considered carefully.

Secure Browser

Safari can be used in conjunction with per-App VPN so that all access to intranet sites is via a VPN tunnel. There are two potential issues with the use of Safari with PROTECTED intranet content:

- Safari caches are stored on device in "ProtectionCompleteUntilFirstUserAuthentication", and ASD requires PROTECTED data to be stored in "ProtectionComplete" or "ProtectionCompleteUnlessOpen".

- If there is a personal Apple ID on the device, and iCloud restrictions allow it, then Safari bookmarks may sync to other devices with the same Apple ID.

Use of private browsing mode helps mitigate both of these issues, but is not enforceable by policy.

In some cases, where PROTECTED data is in use, per-App VPN can be used, with a "Secure Browser App" which may be mapped to use the configured per-App VPN for corporate network traffic exclusively. There are several advantages to using a separate secure browser:

- A secure browser may be chosen that protects cached data and credentials in a higher data protection class than Safari, and is suitable for cached content to be at a PROTECTED classification.

- Depending upon the MDM solution, a secure browser may have corporate data wiped when a device is lost/stolen, out of compliance, or upon other configurable triggers.

- When the secure browser is mapped to the corporate intranet per-App VPN, users may be permitted to use Safari for ordinary internet access. This reduces the security risk of sensitive or classified data transmission from corporate intranet to internet. In this case, Safari may still be configured with a web proxy to protect against web based threats and for acceptable use policy enforcement.

Global Proxy

Since iOS 6, supervised devices may be configured with a "Global Proxy" setting. This allows administrators to configure supervised devices to use a specified HTTP(S) proxy on all network interfaces. Global Proxy is just a convenient way of setting the proxy settings in multiple locations, and only applies if an App developer used high level networking APIs that are aware of network state changes and proxies. If a developer has coded to the CFNetwork layer, or even raw unix sockets, then an App will simply ignore the Global Proxy configuration. Even Apple's own Apps built into iOS are a mixture of NSURLSession and CFNetwork and some are not aware of Global Proxy.

In iOS 9, an alternative that may be preferable is combining AOVPN with a transparent proxy behind the VPN end point. Correlation between VPN and proxy logs will allow per-user accountability. iOS 9 also provides under the Network Extension Framework, a mechanism for content filters to examine all traffic, but solutions currently on the market are focussed on an education market context, and ASD is not in a position to comment on their suitability for agencies at this time.

iCloud

iCloud is Apple's brand name for a wide range of cloud services. This section is intended to assist agencies that are looking to make use of those services. As with the use of any cloud service, irrespective of vendor, agencies must ensure that

their use complies with the Department of Finance's policy and guidance, see

https://www.finance.gov.au/cloud/

Additionally, there are security risks that need to be addressed. For agencies handling UNCLASSIFIED with DLM and PROTECTED data they must ensure they are compliant with the controls in the ISM that address cloud security, in particular the *Outsourced Cloud Services* and *Outsourced General Information Technology Services* sections. At the time of writing, none of the cloud services under the iCloud brand have been awarded ASD certification and iCloud is not an ASD Certified Cloud Service. For security guidance on cloud services and the ISM, see:

http://www.asd.gov.au/infosec/cloudsecurity.htm

http://www.asd.gov.au/infosec/ism/

The cloud services under the iCloud brand and their data are hosted on a mixture of infrastructure controlled by Apple and third parties, such as Microsoft Azure, and Amazon Web Services. When Apple uses third party services, only encrypted data is present on the third party servers, with the key material being a direct transaction between Apple and the device - it is never decrypted on the third party servers, nor do the encryption keys ever transit those servers. Apple's server infrastructure is currently only in the USA, spread across multiple sites, but is expanding slowly internationally (Apple's first non-USA based data centres have been announced in Europe). The security risks associated with iCloud services vary by cloud service as for some cloud services Apple has visibility of data or key material, and for other cloud services it does not. Generally, access to iCloud data is controlled by the iCloud account aspect of an Apple ID - i.e. if 2 factor authentication is not enabled, then any device with that set of credentials present on it can access the iCloud data. It is therefore highly desirable that 2 factor authentication be enabled for ANY Apple ID in use.

This primary exception to data exposure risk with iCloud is backup. An App can be built, or configured by MDM, to be excluded from a backup. This is an extremely important control in mixed use device scenarios, as it can permit user data and configuration to be backed up, but prevent the contents of Apps containing sensitive or classified data from doing so.

Note also that access to iCloud can be implicitly blocked by preventing an iCloud account being added to the device, or explicitly using MDM restrictions.

Table 2.1: iCloud Services

iCloud Service	Apple Can Decrypt	Tied to AppleID	Location	Remarks
iCloud Mail	Yes	Yes	Apple	Normal POP/IMAP mail services
iCloud Contacts	Yes	Yes	Apple	CardDAV
iCloud Calendars	Yes	Yes	Apple	CalDAV
iCloud Drive	No	Yes	Distributed, Apple controls keys	Scope of trust is limited by Apple ID and public keys of trusted devices
iCloud Document and Data Sync	No	Yes	Distributed	Scope of trust is limited by Apple ID and public keys of trusted devices
iCloud Keychain	No	Yes	Distributed	Scope of trust is limited by Apple ID and public keys of trusted devices
iCloud Backup	Yes	Yes	Distributed, Apple controls keys	See Backup Table for more details
iCloud Photo Library	No	Yes	Distributed	Scope of trust is limited by Apple ID and public keys of trusted devices
Siri	Yes	No, Anonymous Identifier used. Capability to infer identify unclear	Usually “onshore”, but can’t control	Siri servers are usually present in countries where a local language dialect is supported, but Apple can not guarantee this data stays “onshore”.

Table 2.2: Device Backup Settings

In iCloud Backup	In Unencrypted iTunes Backup	In Encrypted iTunes Backup
Third Party App Data (if not excluded by device or per-App policy)	Third Party App Data (if not excluded by device or per-App policy)	Third Party App Data (if not excluded by device or per-App policy)
Purchase history from App Store, iBook Store and iTunes Store	Purchase history from App Store, iBook Store and iTunes Store	Purchase history from App Store, iBook Store and iTunes Store
Device settings	Device settings	Device settings
Photos and Videos (if not using iCloud Photo Library)	Photos and Videos (if not using iCloud Photo Library)	Photos and Videos (if not using iCloud Photo Library)
Home screen & folder layout	Home screen & folder layout	Home screen & folder layout
SMS and iMessages saved on device	SMS and iMessages saved on device	SMS and iMessages saved on device
Ringtones	Ringtones	Ringtones
Visual Voicemail Token (note this is paired to the carrier SIM)	Visual Voicemail Token (note this is paired to the carrier SIM)	Visual Voicemail Token (note this is paired to the carrier SIM)
HealthKit configuration	HealthKit configuration	HealthKit configuration and data
HomeKit configuration	HomeKit configuration	HomeKit configuration
Voice memos	Voice memos	Voice memos
		Passwords and Certificate Identity secret keys
		Wi-Fi configuration including secret keys
		VPN configuration including secret keys
		Web browsing history

All content in a backup that is encrypted, is first encrypted by the Hardware Cryptographic Module (HCM) in the device prior to leaving the device. The cloud service or host computer may subsequently super-encrypt that data. Note that iCloud backup only occurs on a daily basis when a device is powered on, locked, connected to Wi-Fi and connected to a power source.

Note that in an iCloud backup: certificates, photos and video, voice memos, and SMS and iMessages are encrypted using AES-256 with the originating device's UID by the HCM, and can not be restored to a different device.

iTunes backups are automatically encrypted if the device has a passcode set, or is managed by MDM. Devices that don't meet these conditions can still be set to be encrypted. If an encrypted iTunes backup is created, all encryption/decryption is done on device using the HCM.

If an agency decides that iCloud Backup is permitted, then:

- Apps containing PROTECTED or UNCLASSIFIED with DLM data should be excluded from backup by either MDM policy or enterprise Apps self-excluding.
- Photo and video content that is PROTECTED, should be using dedicated Apps, not the system provided Camera and Photo Reel.
- iMessage should be considered for UNCLASSIFIED only.
- It may be desirable to enable a local backup mechanism using agency controlled computers running iTunes to facilitate users migrating between devices.

For further information about backups on iOS, see:

https://support.apple.com/en-au/HT204136

https://support.apple.com/en-us/HT205220

https://support.apple.com/en-au/HT203977

Continuity

Continuity is the name given to a group of iOS and Mac OS X features which enhance device to device collaboration. Continuity encryption identities are based on the Apple ID - i.e. if enabled, it shares data with other devices with the same AppleID configured that are in close physical proximity. For more information on this feature visit Apple's website:

https://www.apple.com/au/ios/whats-new/continuity/

Continuity offers some productivity benefits for agencies, while at the same time creating some new concerns such as:

- A corporate iOS device may handoff a sensitive or classified document being worked on to a user's personal Mac with the same Apple ID. Handoff may also be administratively disabled by Configuration Profile restriction.
- For phone calls and messages, the level of protection given to data on an untrusted Wi-Fi network. Refer to Apple's *iOS Security Guide* for details.

App Extensions

Beginning with iOS 8, third party iOS App developers may create functionality which may be utilised by other Apps on the device. App Extensions cannot be disabled through Configuration Profile restriction. For details regarding features, please refer to Apple's documentation:

https://developer.apple.com/app-extensions/

Some agency concerns with App extensions may include:

- a user installs custom keyboard extension which transmits keystrokes to a third party
- a user uploads sensitive or classified data to a third party document provider
- a user shares sensitive or classified data using a custom share extension.

When an App with an App Extension is installed, both the App and the App Extension may be considered managed or unmanaged depending upon how the App was installed. Additionally, App Extensions respect Managed Document Configuration Profile restrictions.

If the following settings are disabled by Configuration Profile they have the following effects:

Allow documents from Managed Apps in Unmanaged Apps

App Extensions installed with MDM deployed Managed Apps can't be activated by user installed App Store Apps. For example, a corporate installed document provider extension can't be used by a user's personal App Store App.

Allow documents from Unmanaged Apps in Managed Apps

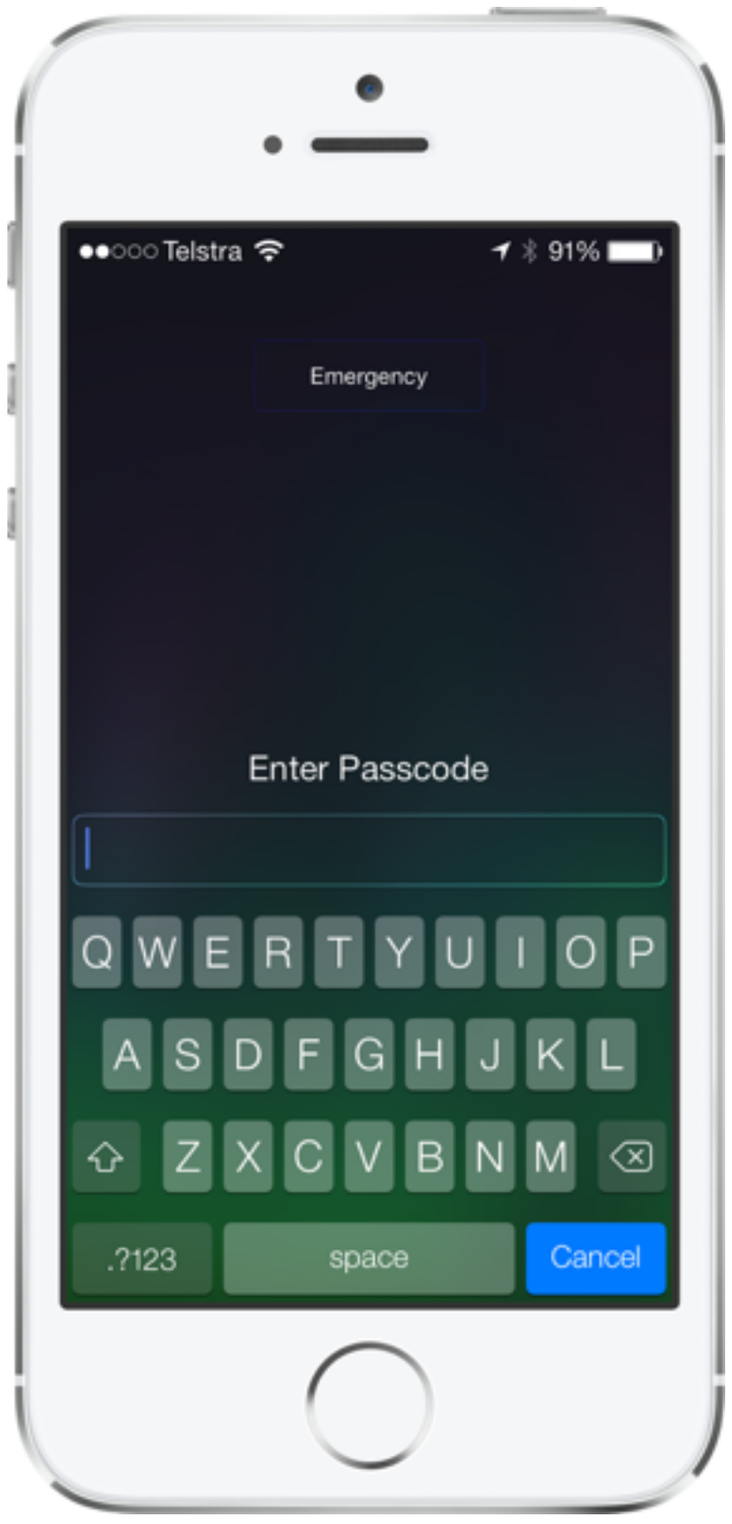
App Extensions installed by user App Store Apps can't be activated by MDM installed Managed Apps. For example, a user installed custom keyboard can't be used by corporate Apps.

Capability	Enablers	Comment
Remote Device Wipe	MDM & APNS (preferred) EAS (optional)	Remote wipe can be initiated by an administrator through MDM. Some MDMs also allow users to initiate. Third party MDM software may offer a separate “Enterprise Wipe” which erases data in Managed Apps.
Proxy	VPN with Transparent Proxy (preferred) Global Proxy	A proxy can be set on a VPN session.
Firewall	VPN to firewalled corporate network enclave	iOS does not implement a local firewall. This is significantly mitigated by the hardened runtime environment and use of Always On IPSec IKEv2 VPN.
Force Device Settings	MDM (preferred) Apple Configurator 2	Apple Configurator 2 and MDM may be used to generate, sign and deploy Configuration Profiles to iOS devices.
Multi-factor Authentication	TLS CA infrastructure, DNS, RSA or CryptoCard (VPN Only), Smartcard (requires third party software and hardware)	Depending on the agency’s security posture, device certificates or soft tokens may be considered as a second factor of authentication. At the time of writing, Touch ID can not be used for multi-factor authentication.
OTA Configuration Profile (pull)	TLS CA infrastructure, DNS, Web Service, Directory Service	Sign profiles using enterprise CA infrastructure.
Mobile Device Management	Enterprise Developer Agreement, 3rd Party MDM appliance, CA infrastructure, DNS, Directory Services, APNS	MDM should be tied into CA and Directory Services.
Remote Application Deployment	Enterprise Developer Agreement, Web Server, 3rd Party MDM appliance (optional), APNS (optional)	Enterprise and App Store Apps can be remotely deployed. OS X Server Caching Server may help reduce corporate network internet consumption.
Home screen	Apple Configurator 2 / MDM	Set Home screen to “If found return to PO BOX XXXX”.

Table 2.3: Capability enablers

CHAPTER 3

Encryption in iOS



Keeping data safe when the device is out of your control.

Encryption in iOS

This chapter is provided to help agencies understand the underlying encryption architecture employed in iOS 9 to assist them to make an informed assessment of the security risks to Australian government information. Apple documents the security architecture of iOS and associated services, including the encryption architecture, in its iOS Security Guide, available from:

https://www.apple.com/business/docs/iOS_Security_Guide.pdf

Data protection

Apple has explained that one of the goals of iOS is to 'keep data safe even when the device is compromised'. In all recent versions of iOS, all objects on the File System are encrypted all the time, using the HCM. What changes when a passcode is set, is the policy that is applied by the kernel for access control to the encryption keys for objects. As will be explained in this chapter, significant onus remains on App developers as to how data protection policy is applied. For this reason, it is important that an agency wishing to use a particular App understands the security features of iOS 9, and how the App uses them, in order to make a more informed decision as to whether the App meets the security needs of the agency.

It is important to note that if devices are not configured to use the Managed Open-In feature, users can still move files to Apps that use lower data protection classes.

In general, data classified as PROTECTED should be stored at rest in Class A or Class B data protection classes whenever possible. In configurations where there is limited personal use of the device, PROTECTED data should be managed i.e. in Apps that were installed by MDM, and are subject to additional policy controls.

The iOS 9 default data protection class is Class C, which is appropriate for UNCLASSIFIED or UNCLASSIFIED with DLM.

Secrets and Data

Within iOS 8 and later, information stored by Apps can be broadly categorised as either a secret or data. The term secret can be understood to mean information by which one may get access to data; this can include system credentials, keys and passwords. Data on the other hand, refers to user/application data such as text, pictures and documents. Accordingly there are two data stores where a developer should choose to store information: the Keychain and the File System. Developers are encouraged to store secrets within the Keychain and place more general application data within the File System.

Information stored within the Keychain or File System can be customised to different levels of security, accessibility and portability. Note that it is entirely up to the developer to determine the level of protection applied. This choice is made by the App developer through API calls and the choice of availability as detailed in Table 3.1. If a developer takes no specific action, an App defaults to “After first unlock” behaviour. “When unlocked” behaviour can be forced by a checkbox setting at compile time, and does not require specific coding. If

an App needs to sync data in the background, or change the data protection class of an object, then a developer needs to write code for this.

The default behaviour for most iOS Apps, is that the files utilise "...CompleteUntilFirstUserAuthentication", and most iOS Keychain items utilise "…AfterFirstUnlock".

Note: Agencies developing or making use of Apps handling sensitive or classified data should take care to investigate how data is handled within their App. They must ensure the appropriate data stores and availability flags (outlined in Table 3.1) are used to achieve the secure handling of Australian government information.

Classes of Protection

An App developer has the option of setting the following availability flags with any File System Element or Keychain entry they create.

Availability	File System Element	Keychain Entry
When unlocked	...Complete	...WhenUnlocked
While locked	...CompleteUnlessOpen	N/A
After first unlock	...CompleteUntilFirstUserAuthentication	...AfterFirstUnlock
Always	...None	...Always

Table 3.1: File system class accessibility

From Table 3.1, it is possible to abstract these settings into four standard classes of containers with the following behaviour:

Class A: Files and credentials within this data protection class can only be read or written when the device is unlocked.

Class B: Through the use of public key cryptography, files within this data protection class can be written after the device is initially unlocked, and can be read only when unlocked.

Class C: Files and credentials within this data protection class can be read or written only after the device is initially unlocked. App Store Apps use this data protection class by default. Powering off or rebooting the device will render data in this protection class inaccessible.

Class D: The lowest protection class, files and credentials within this data protection class can be read or written to in all conditions. This data protection class is deprecated in iOS 9, and may go away completely in future versions of iOS.

Note that there are other attributes beyond these data protection classes available to developers who wish to take advantage of them.

- "this device only" wraps the object with the device key, so it is not portable to other devices. Almost all credentials stored on the device set this attribute.
- "when passcode set, this device only" is available to a subset of Class A objects, which become unavailable if the passcode is removed from the device.
- "DeviceOwnerAuthenticationWithBiometrics" is available to keychain objects, and is only accessible while unlocked, while a passcode is set and following successful TouchID authentication. This class is very useful for storing credentials used to authenticate Apps or services, in conjunction with ProtectionComplete

iOS encryption architecture

iOS devices with an A7 or later processor include a "Secure Enclave". This is a separate processor that runs an L3 kernel that serves as a hardware keystore (wrapping all Keychain data, and data protection class keys), and has its own hardware random number generator (RNG). In addition, in iOS 9, the Secure Enclave can be used to generate public/private key pairs, and only ever expose the public key to iOS.

Data protection classes are shown in Figure 3.1 which illustrates an example where four files exist, each assigned a different data protection class:

- File 1 is of type Class A: accessible only when unlocked
- File 2 is of type Class B: can be written to after first unlock, but can only be read when unlocked
- File 3 is of type Class C: accessible after first unlock
- File 4 is of type Class D: always accessible.

Note that while files were used for the purposes of this example, with the exception of the Class B data protection class, Keychain entries could just as easily be used in their place.

Similar to the File System, an App's credentials stored within the Keychain are encrypted using the appropriate Class Key found within the System Keybag (refer to the *Keybag Section* of this chapter for more information).

However, as illustrated in Table 3.1, the protection offered by the Class B data protection class is only available to File System Elements.

In Figure 3.1, irrespective of class, each file is encrypted with both a unique File Key and a File System Key.

The File System Key is used to encrypt all data within the device. As it is stored openly its use does not add to the cryptographic security of data, but is instead used to facilitate a

remote wipe. Refer to the *Remote Wipe* section of this chapter for more information regarding this function.

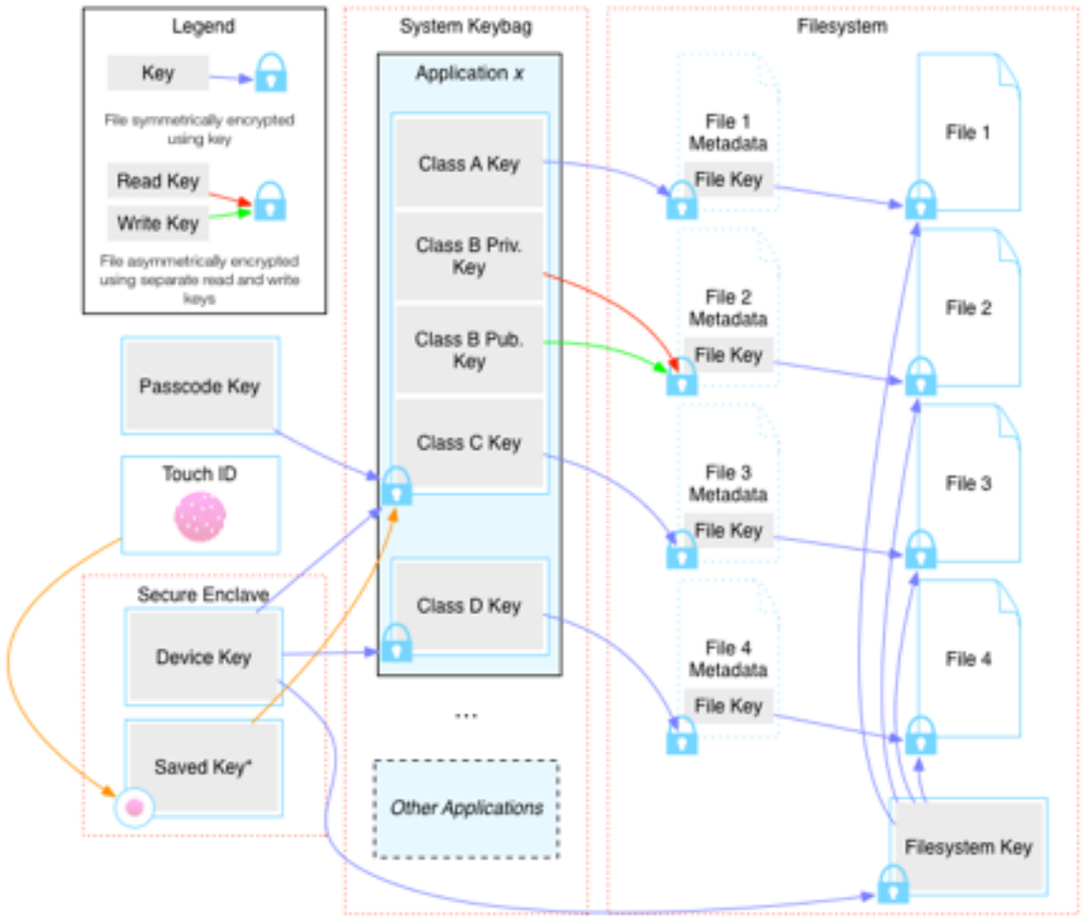


Figure 3.1: iOS File System Architecture

The File Key is stored within the file's metadata, which is itself encrypted by the file's corresponding Class Key. The System Keybag stores all Class Keys within the device. Refer to the *Keybag* section of this chapter for more information on different types of Keybags used throughout the system.

Upon turning on the device, the Class A, Class B (public and private) and Class C keys are initially inaccessible as they rely on the Passcode Key to be unencrypted.

When the device is first unlocked by the user, through the use of their passcode, these keys are unencrypted, stored for use and the derived Passcode Key promptly forgotten.

The Device Key is stored within, and never divulged from, the Hardware Security Module (HSM). This acts to encrypt and decrypt files at will using the Device Key. Refer to the *Hardware Cryptographic Module* section of this chapter for more information on this component.

The Class D Key is encrypted using the Device Key. As this decryption process is always available, irrespective of the state of the device, files protected by this Class Key are always accessible.

Finally, when the device re-enters a locked state, the Class A Key and Class B Private Key are forgotten, protecting that data, leaving the Class C Key and Class B Public Key accessible.

Remote wipe

Remote wipe is the ability for a network connected iOS device to have the data within the device made inaccessible (enacted by receiving a system command). This is achieved in iOS by erasing the File System Key, which is used by the device to encrypt all user data (as shown in Figure 3.1). For this reason, once this key is removed, no user data on the device is retrievable.

Hardware Cryptographic Module (HCM)

Internal to the device, the HCM is the only means by which iOS can make use of the Device Key. This Device Key is unique to the device and is not exportable using any non-invasive technique. The Secure Enclave and the HCM establish trust and communicate via the Application Processor.

For this reason (as files encrypted with the Device Key can only be decrypted on the device), the iOS architecture makes itself resistant to offline attacks. The most significant being a brute-force attack to exhaust and thus discover the user's Passcode Key. All such brute-force attempts rely upon the HSM, are performed on device and are rate limited by iOS on devices with A6 and earlier processors, and in the Secure Enclave by A7 and later devices.

Note: The Secure Enclave and HSM have both been evaluated under FIPS-140-2 and Common Criteria's *Mobile Device Foundation Protection Profile Version 2.0*.

Keybags

There are three types of Keybags used in iOS: System, Backup and Escrow.

All Keybags are responsible for storing the systems Class Keys, which are in turn used to gain access to individual files or Keychain entries (as shown in Figure 3.1).

The System Keybag, shown in Figure 3.1, is used internally within the device to facilitate the user's access to the File System and Keychain.

The Backup Keybag is designed to facilitate backups in a secure manner. This is done by transferring the encrypted contents of the File System, and Keychain to a remote system along with the Backup Keybag.

The user then has the option to password protect this Keybag; this decision has implications concerning the portability of the Keybag. If the user specifies a password, the Backup Keybag is then encrypted with this password.

Given the password, this data can then be restored to an iOS device (Note: if a developer has specified data 'ThisDeviceOnly', such data will not be made portable). If the user does not set a password, then the Backup Keybag is protected using the Device Key which never leaves the device. Consequently, the Backup Keybag can only be restored to the original device.

The Escrow Keybag is designed to enable a paired device (normally a computer) to gain full access to the device's File System when the device is in a locked state. In this context

pairing refers to connecting the iOS device in an unlocked state (or within 10 seconds of being in an unlocked state) to the other device in question.

An exchange then occurs, where the paired device receives a copy of the iOS device's Escrow Keybag. This Keybag is encrypted using the iOS device's Device Key, thus restricting access when disconnected from the iOS device.

Setting a passcode

Setting a passcode is required to enable data protection. In most environments enabling a passcode will form part of agency policy, and this will be enforced either over EAS, or via a Configuration Profile installed on the device. For passcode policies see *Suggested Policies* in Chapter 6.

Verifying data protection is enabled

There are two main methods of verifying that the File System of a device has been configured to support data protection. An MDM console can query the data protection status and report centrally. The user of a device can also validate if data protection is enabled by navigating to Settings > General > Passcode Lock and scrolling to the bottom of the screen. If data protection is enabled, "Data protection is enabled" will be displayed at the bottom of the screen.

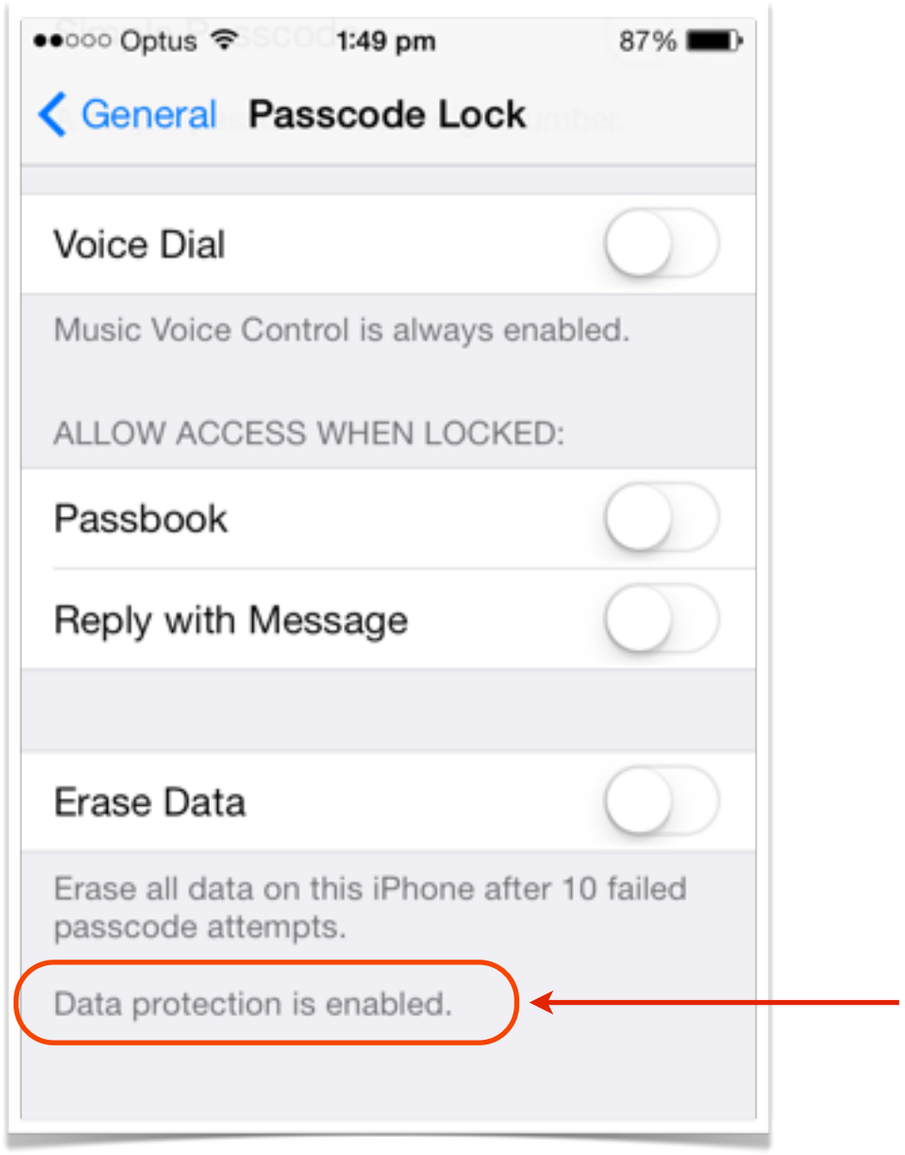


Figure 3.2: Data protection enabled

References and further reading

For more information on the encryption used in iOS, please refer to the following:

iOS Security May 2016: https://www.apple.com/business/docs/iOS_Security_Guide.pdf

iPhone data protection in depth by Jean-Baptiste Bedrune and Jean Sigwald from Sogeti

iOS filesystem decryption tools and information from Bedrune and Sigwald: http://code.google.com/p/iphone-dataprotection/

Apple iOS 4 Security Evaluation by Dino A. Dai Zovi

Session 709: Protecting Secrets with the Keychain, Apple Developer WWDC 2013 Presentation

Session 208: Securing iOS Applications, Apple Developer WWDC 2011 Presentation

CHAPTER 4

Deploying iOS Devices

Enrol and sanitise iOS devices using Apple Configurator 2.

Deploying iOS devices

iOS has a number of features that are aimed at helping administrators manage iOS devices in large agencies. In most cases, agencies must use a combination of available tools during deployment.

Apple provides general purpose guidance on management and deployment of iOS devices in its *iOS 9 Deployment Reference*:

https://itunes.apple.com/au/book/ios-deployment-reference/id917468024?mt=11

There are two main deployment options:

- Over the air, which typically uses MDM, for both BYOD and agency owned deployments. It may use Device Enrolment Program (DEP) for agency owned devices, to force MDM enrolment and make it non removable.
- Tethered, which typically uses Apple Configurator 2. It may also optionally incorporate MDM and DEP. It is most useful for shared devices that do not require active monitoring and frequently return to a known location to be managed, such as in a training environment, or devices that rarely connect to a network.

Apple Configurator 2

Apple Configurator 2 is a OS X application that allows administrators to set up and deploy groups of similarly configured iOS devices. Apple Configurator 2 may be suitable for:

- preparing devices for PROTECTED deployments
- preparing devices for MDM enrolment
- activating and naming groups of new devices
- updating iOS on groups of devices
- installing Apps and Configuration Profiles on groups of devices
- backing up and restoring devices
- saving and retrieving documents.

If an agency is not able to use DEP it is recommended that administrators use Apple Configurator 2 to supervise devices, in conjunction with an MDM to manage over the air, for large deployments of iOS devices.

Supervised mode

A key feature of Apple Configurator 2 and DEP (linked to MDM) is the ability to place devices into what is called “supervised mode”. Supervised mode flags the device to the policy engine built into the operating system as being owned by an agency, and allows a greater range of restrictions to modify standard iOS behaviour. Some of the effects include:

- devices are able to be administered using Configuration Profile restrictions not normally available
- devices can be prevented from syncing music or media from computers running iTunes
- devices can be prevented from connecting to or installing Apps using iTunes
- devices can be prevented from being backed up using iTunes
- significantly increased difficulty in jailbreaking, even with full co-operation of the device user
- in some cases, users not being notified when changes are made to device configuration
- administrative message on lock screen
- Activation Lock can be managed by MDM
- device data being erased upon initial provisioning.

Though it is not usually appropriate to use supervised mode in a BYOD model, there are reasons why supervised mode is desirable for agency owned devices:

- Sensitive or classified data on each device is better protected. Users cannot sync or backup their device contents to their home computer. iOS forensic recovery utilities may not be able to recover data from the device without a jailbreak.
- Users cannot easily sidestep restrictions without erasing the device.
- Supervised mode increases the difficulty of a number of attacks that rely upon the USB host pairing protocol.

Devices not configured as supervised devices are referred to as unsupervised devices.

Note: PROTECTED devices must use supervised mode. Agency owned devices should be placed in supervised mode.

Supervisory Host Identity Certificate

Normally, an unlocked iOS device is able to pair with any host running iTunes (or supporting the lockdown protocol). When an iOS device is set to supervised mode, it authenticates with a host using the “Supervisory Host Identity Certificate”. If the “Allow pairing with non-Configurator hosts” Configuration Profile restriction is disabled, a device will only pair with a host running Apple Configurator 2 with the correct Supervisory Host Identity Certificate. Ordinary pairing with iTunes is not possible with any other hosts. On a Mac host running Apple Configurator 2, the Supervisory Host Identity Certificate is stored in the login Keychain, and can be manually exported.

For UNCLASSIFIED deployments, note that Apple Configurator 2 can be used to sync the supervision identity, and Apple Configurator settings, via iCloud to other instances of Apple Configurator logged in with the same Apple ID. This would permit multiple Apple Configurator 2 workstations at multiple locations to have the same supervision identity. It can also be synced using a manual export/import process, if required. This method is not appropriate for higher classifications at this time.

While supervised devices with this restriction are unable to establish new trust relationships with iTunes hosts, a trust relationship will be formed between devices and the Apple Configurator 2 host. A record of this trust relationship is stored in Escrow Keybag files, which on OS X are located at: /var/db/lockdown. It is important to ensure that the Apple Configurator 2 host is regularly backed up, loss of the Supervisory Host Identity Certificate and Escrow Keybag files will mean that supervised devices cannot be administered via the USB interface in the future (unless the devices are wiped and re-provisioned with a new identity).

Note: Escrow Keybag files and exported Supervisory Host Identity Certificates should be protected in a similar manner to private keys.

Activating devices with Apple Configurator 2

Apple Configurator 2 will attempt to automatically activate all connected devices after operating system installation. It is important for administrators to note that iPhones and some iPads require a SIM for activation. If the SIM has a passcode lock, automatic activation will be unsuccessful.

Installing iOS

A key feature of Apple Configurator 2 is its ability to install iOS on many devices concurrently. Additionally, varied device platforms (iPhone, iPad and iPod) can all be simultaneously connected. Apple Configurator 2 will seamlessly download iOS for all supported device platforms when there is an internet connection available. A theoretical maximum of 64 devices can be connected concurrently for installation.

Installing Configuration Profiles

Apple Configurator 2 may be used both to install Configuration Profiles and to create new Configuration Profiles. These profiles can be installed on devices in bulk when initially preparing devices for deployment. As an example, this may be used to initially roll out a trust profile for an agency MDM server.

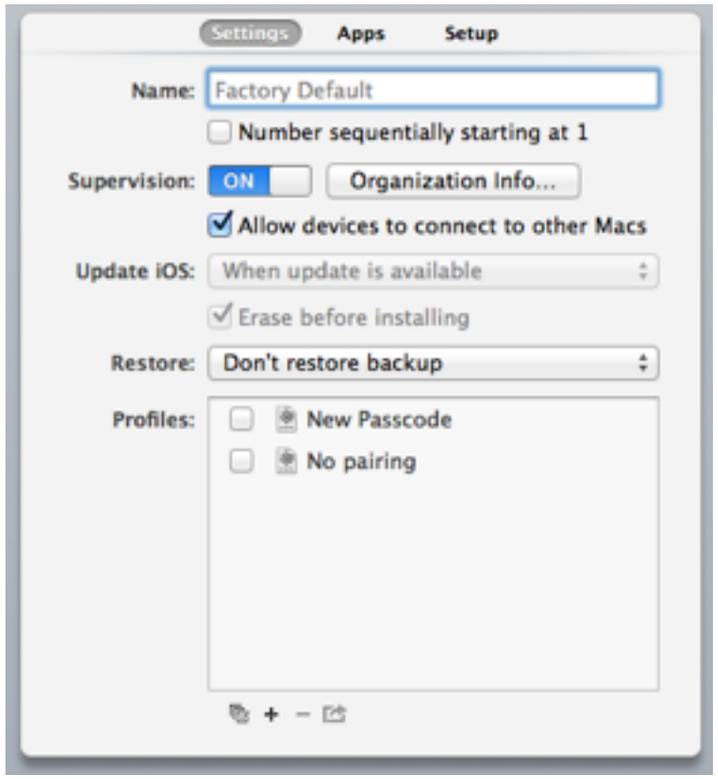


Figure 4.1 Installing Configuration Profiles with Apple Configurator 2

iOS Updates

There are two methods for Apple Configurator 2 deployed devices to receive iOS updates. Devices with an internet connection will prompt users to install over the air updates. Alternatively, users can return their devices to their administrator to have them updated using Apple Configurator 2.

References and further reading

Refer to the following publication for additional information on Apple Configurator 2:

http://help.apple.com/configure/mac/2.2.1/

Device Enrolment Program

The Device Enrolment Program (DEP) is a deployment feature which allows agencies to pre-configure iOS devices, allowing these devices to be delivered directly to the user.

Devices can be preset to supervised mode and can be forced to connect to an agency's MDM system. DEP is the only mechanism to make MDM non-removable, and is therefore strongly recommended for most deployment scenarios. There are numerous benefits for administrators in deploying a fleet using DEP:

- save time deploying by not having to physically connect devices to host running Apple Configurator 2
- no need to use Apple Configurator 2 to re-provision a device after the device is wiped
- devices can be re-deployed to new users in the field
- lost/stolen devices may be recovered or rendered useless
- MDM enrolment can be made mandatory and non-removable.

The administrator Apple ID used to control the DEP for an organisation has considerable power. Apple requires such AppleIDs to use two-factor authentication. Agencies should

conduct a risk analysis on the use of this feature. At a minimum they must consider the protocols they put in place to maintain control over this Apple ID.

Activation Lock

Activation Lock is a feature which requires Apple ID (iCloud) user authentication before device activation. Typically Activation Lock is enabled if a user enables iCloud > Find my iPhone in the Settings App on an unsupervised device.

If a user has enabled Activation Lock on an iOS device, that user's Apple ID will be required to activate the iOS device after (or during) sanitisation, unless the device is supervised, and managed by MDM (in which case the MDM can clear Activation Lock). For this reason it is recommended that agencies should supervise all agency owned devices, and should use MDM to manage Activation Lock.

Device Sanitisation

Administrators should erase and re-provision devices for the following reasons:

1. to sanitise a returned iOS device for re-issue or disposal
2. to sanitise an employee owned iOS device before provisioning
3. to sanitise an employee owned iOS device prior to the employee being deployed
4. to sanitise an employee owned iOS device prior to the employee leaving the agency
5. to break all device-to-host trust relationships and invalidate old Escrow Keybag files.

Breaking the device-to-host trust relationship

When an iOS device pairs with a host, a trust relationship is formed. If there is a need to remove trust relationships, this can be done on device by going to Settings -> General -> Reset. In many cases an administrator may want to erase an iOS device and break all the established device-to-host trust relationships that a device has previously created. The most reliable method to break all established relationships is to restore the iOS device firmware using what is commonly known as "Device Firmware Upgrade" mode (DFU mode).

Note: Restoring a device in this way will also erase all data and settings on the device.

The DFU mode restoration can be performed from a host that has no established trust relationship with the device, and the device passcode is not required.

DFU Mode Restoration

To perform an iOS firmware restoration follow this procedure:

1. connect the iOS device to the host PC or Mac running iTunes
2. if iTunes is unable to pair with the iOS device, please clear any error dialog boxes
3. press and hold both the Sleep/Wake and Home buttons on the iOS device for ten seconds
4. release the Sleep/Wake button, and continue to hold the home button
5. release the home button after iTunes generates the following dialog box

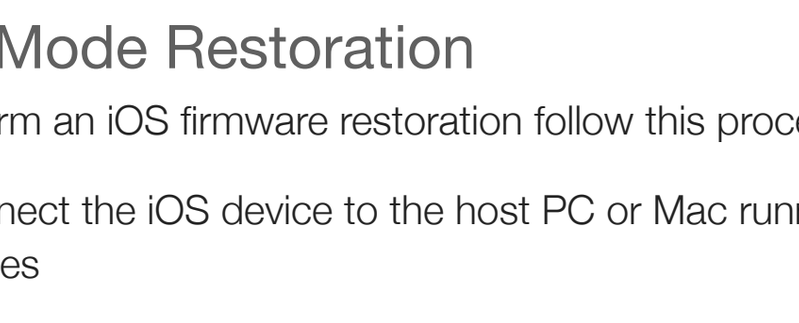


Figure 4.2: iTunes detects device in recovery mode.

6. after clicking OK, click the Restore button

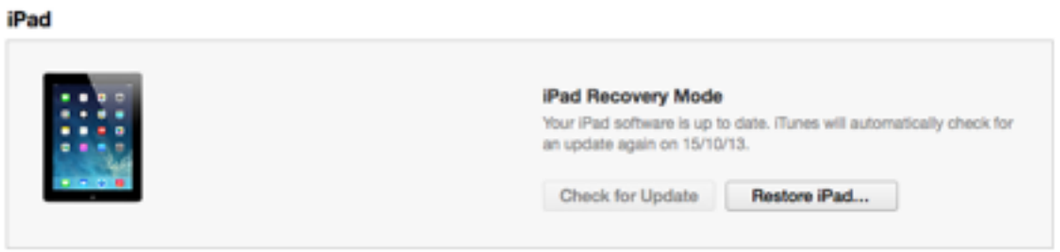


Figure 4.3: Restore iPad.

7. begin the iOS restoration process by clicking Restore and Update.

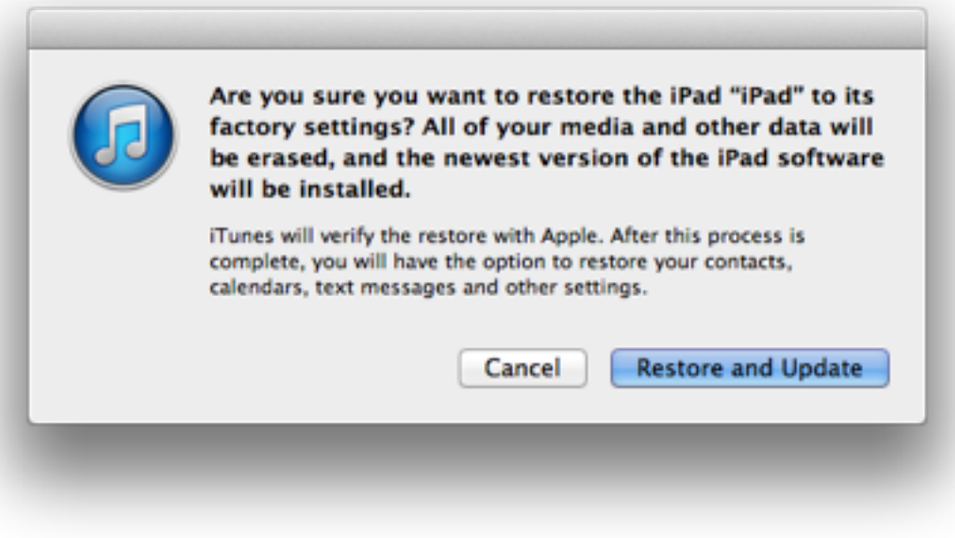


Figure 4.4: Restore and Update.

Sanitising an iOS device for re-issue or disposal before provisioning

If an agency owned device is returned for re-issue to another employee, or disposal, it should be erased by performing a DFU mode restoration. One reason that this is important is so that the previous iOS device owner cannot take advantage of any old device-to-host trust relationships to retrieve data from the device. By performing the DFU mode restoration the old trust relationships are broken.

Before an iOS device is re-provisioned for enterprise use, it is recommended to perform a DFU mode restoration. This will ensure that the device is in a known state.

Sanitising employee owned iOS devices

Employee owned iOS devices are making their way into agencies, creating a new set of challenges for administrators. Some of the challenges being faced include:

- jailbroken devices running untrusted code
- jailbroken devices being able to bypass all security protection (including third party managed containers)
- unpatched iOS devices running old iOS versions that are vulnerable to exploitation
- devices previously configured with conflicting settings and/or Configuration Profiles.

Older versions of iOS

Each revision of iOS includes many security related fixes. If left unpatched, iOS devices could be exploited remotely, risking both employees' personal information and the security of the agency network.

The agency acceptable use policy should require users to install iOS updates as they become available.

MDM can monitor the iOS version installed on devices, and either:

- notify users managers that a staff member has not updated
- prompt the use to update the device.

Jailbroken employee owned devices

Jailbroken devices allow users to install applications on their iOS devices from outside of Apple's App Store. Though there are many useful legal purposes for jailbreaking a device, jailbreaking carries with it a number of negative side effects that impact the security of the agency network and the confidentiality of data stored on a device:

- Jailbreaking usually disables App code signing checks. These iOS code signing checks help to prevent malware executing on a device. Removing this makes exploitation easier.
- Jailbreaking may disable or break important security features such as Address Space Layout Randomisation (ASLR) and application sandboxing. ASLR increases the difficulty of successful exploitation of a vulnerability. Malware on a jailbroken device would not be constrained by the application sandbox.
- Jailbroken devices often have serious unpatched operating system vulnerabilities and are more vulnerable to exploitation.
- Publicly available jailbreaks may contain malware.
- Pirate App Stores that cater to Jailbroken devices have very high levels of malware.
- Jailbroken devices should be assumed to be untrusted.

Jailbroken devices are also often able to enrol into MDM systems without detection. Indeed, there are Apps that have been written with the purpose of evading MDM agent jailbreak detection. When this occurs, administrators can not have confidence that Configuration Profile restrictions and settings are enforced by the operating system.

Information about a commonly used jailbreak detection bypass utility is available at:

http://theiphonewiki.com/wiki/XCon

Administrators should not allow employee owned jailbroken iOS devices to be provisioned for any purpose.

For all the above reasons it is important to ensure that devices are sanitised prior to deployment.

Sanitisation prior to deployment

When considering how to sanitise employee owned iOS devices for enterprise deployment, it is important to take into account the data that employees already have on their devices. Employees may have expectations about how they will be able to use their devices and the effect of enterprise deployment on their device. As an example, an employee might expect their iPhone's contact list to be preserved after deployment. If the device is erased using DFU mode this will not be the case.

If an employee's personal data is to be preserved, the following procedure may be performed prior to enterprise deployment:

1. take a backup of the device
2. perform DFU mode restore
3. restore a backup to the device
4. delete backup from the host
5. provision and deploy the device as per MDM instructions.

This will clear the existing device-to-host trust relationships on the device, but will preserve the employee's data. When following this procedure agencies must consider their legal responsibilities to protect the privacy of their users' data.

If there is no need to preserve an employee's personal data on a device, agencies should simply perform a DFU mode restore.

Sanitisation for departing employees

When an employee departs an agency or no longer requires an iOS device connection to the agency network or computers, it is important to remove existing device-to-host trust relationships from the employee's iOS device. On return, agency owned devices should be sanitised by performing a DFU mode restore as described previously. Employees should be made aware that agency owned devices will be sanitised upon return.

In a BYOD model, the following procedure is suggested for departing employees:

1. remove MDM profile from the iOS device, which will:
   - remove the corporate mail account installed by the MDM
   - remove any Apps which have been installed by the MDM which will also remove any associated data

   This can usually be accomplished simply by un-enrolling the device from MDM at the MDM console.
2. take a backup of the device
3. perform DFU mode restore
4. restore backup to the device
5. erase backup files from the host
6. if the user's Apple ID is associated with a corporate SIM/ phone number, it is also necessary to de-register iMessage (https://selfsolve.apple.com/deregister-imessage).

This procedure will also remove any trust relationships established between the iOS device and any trusted host computers. If the employee does not return their iOS device prior to departing, it may be necessary to use the MDM remote wipe function. Employees should be made aware of this fact in an agency's acceptable use policy.

Removing Trust Relationships

It is possible to remove only the device-to-host trust relationships from a device without performing a complete device erase. This can be accomplished by a device user performing the following procedure:

1. open the Settings app

2. select “General” and then “Reset”

3. select either:

    A. “Reset Location & Privacy” which will also restore all privacy settings to factory default, or

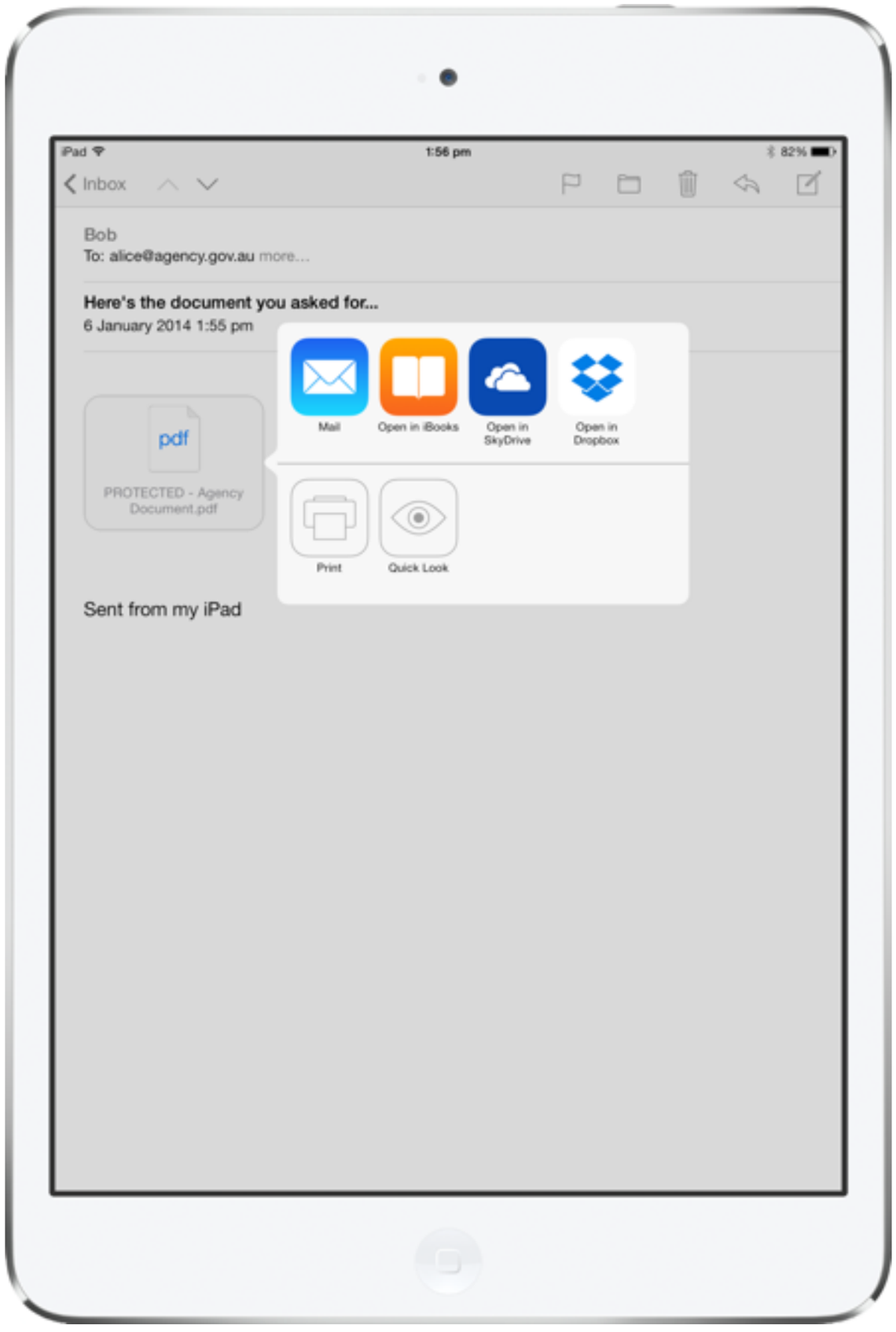
    B. “Reset Network Settings” which will also remove Wi-Fi networks and passwords, Bluetooth pairing records, VPN settings, APN settings and Host pairing records.

For more information on this process refer to:

https://support.apple.com/kb/HT5868

CHAPTER 5

Managing Apps and Data



Understand and mitigate the security risks posed by installing third party Apps.

Managing Apps and Data

Installing iOS Apps can expand the attack surface of an iOS device and increase the probability of a leak of sensitive or classified data. This chapter aims to explain the security risks and provide mitigations. Apple provides developer level technical references for much of this chapter at:

https://developer.apple.com/enterprise/

Security scenarios

Devices running iOS share many of the same security weaknesses of PCs while at the same time presenting some new challenges. Presented below are a few common scenarios that may lead to a leak of sensitive or classified data.

Lost or stolen device

Due to the portable nature of iOS devices, there is always a chance that a deployed device will one day be lost or stolen.

In many situations remote wipe may not be the most desirable option, in some situations it will be impossible. In these situations, adequate data-at-rest protection is vital. When an iOS device is configured following the recommendations given in this guide, data-at-rest protection depends upon how third party App developers have implemented iOS data protection for files and if they have made correct use of the Keychain for credentials.

For a hypothetical security scenario we set the following conditions:

- an iOS device is able to be jailbroken
- has data protection enabled with alphanumeric passcode
- has not been put in to supervised mode
- has third party Apps installed
- has been lost or stolen.

If a third party App developer used the Class D (*NSFileProtectionNone)* data protection class, files within this App's sandbox could potentially be recovered using free or commercial data recovery tools. As an example, if an instant messaging client did not utilise data protection appropriately, chat logs and received files may be recoverable. If the developer stored credentials in the File System rather than using the Keychain, this could mean that the username and password for this instant messaging service may also be recoverable.

Even when data protection and the Keychain are used in an App, they can easily be implemented incorrectly. For example:

- some files may have an inappropriate data protection class
- existing files from an older version of the App have not had their data protection class upgraded
- an incorrect Keychain or data protection class may have been used.

Sensitive or classified information leak

There are a few common ways which Apps may leak sensitive or classified information.

- Apps may allow a file to be opened in another App with inappropriate data protection class or other undesirable behaviour. The second App may for example sync to a cloud file service, or present a file transfer interface to a USB connection.
- Apps may transmit sensitive or classified information over a network with inadequate encryption.
- As part of normal App behaviour, transmit sensitive or classified information over a network to external servers.

In the first case, a good example of an App that shows this behaviour in its default configuration is the iOS Mail App. Using Mail, without any additional policy set by MDM, it is usually possible to open an attachment in another App. For example, an attached PDF file may be opened in the iBooks App. If more than one App has registered a particular file type, Mail will allow the user to choose which App they would like to open the file in. Since the Dropbox App also registers itself as being able to open PDF files, if the user has both iBooks and Dropbox installed Mail will allow a user to open a PDF document in their chosen App, as shown in Figure 5.1.

Although Mail has been approved by ASD for use up to PROTECTED, if an administrator has not configured policy on the device correctly, and if a user chooses to open an attachment in another App it may not be subject to adequate data protection. Additionally, many Apps (including iBooks and Dropbox) may sync files and metadata to cloud services that are not certified to protect sensitive or classified information. In this example, the administrator should apply a restriction policy to “Only allow managed documents to open in Managed Apps”, and this would prevent a mail attachment from being opened in a user-installed App that was unsuitable for PROTECTED data.

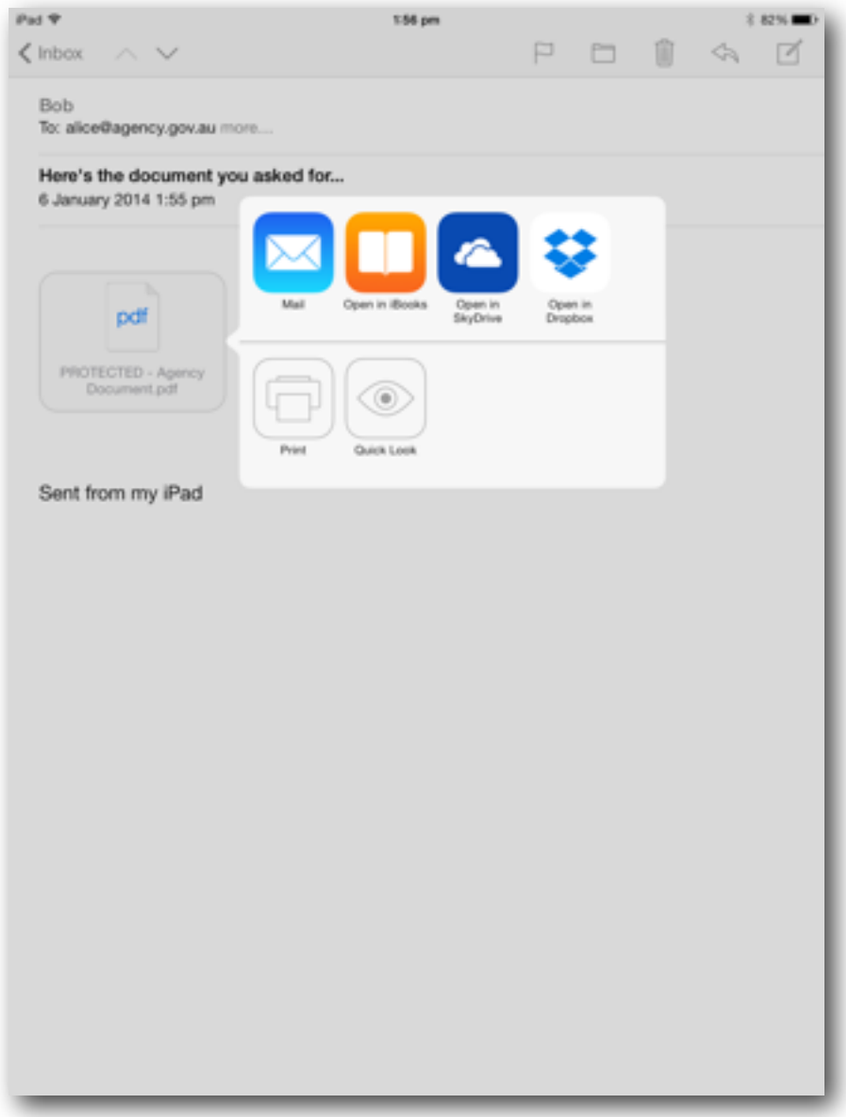


Figure 5.1: Mail app Open-In

Many iOS Apps transmit information to other devices on a network or on the Internet. Often the information transmitted is not sensitive or classified; however there have been incidents where private, sensitive or classified data has been transmitted over the Internet without encryption. For UNCLASSIFIED DLM systems, the ISM (Control 1162) specifies that: “Agencies must use an encryption product that implements an AACP if they wish to communicate sensitive information over public network infrastructure.” For PROTECTED systems, the ISM (control 0465) specifies that: “Agencies must use a Common Criteria-evaluated encryption product that has completed an ACE if they wish to communicate classified information over public network infrastructure.”

Where agencies are utilising On Demand VPN, it is possible that a user may deliberately or accidentally disable the VPN tunnel. In this case, the data’s transport security is limited to whatever scheme the app developer has implemented.

Lastly, there are security risks which cannot be mitigated via iOS technical controls alone:

- copy/pasting sensitive or classified data
- printing sensitive or classified data to the wrong printer
- photographing the screen of the device.

Note that there is no system wide policy for cut and paste control. While deliberate acts cannot in general be prevented, the iOS user interface for cut and paste makes it relatively low in probability this would occur by accident. Policy and user education can further mitigate the case where any of the above events occur accidentally. Deliberate misuse of copy/paste and printing can be managed using a combination of careful custom App development, negotiation with App Store App developers and in some cases, modification of existing Apps using third party add ons.

When developing custom Apps, if copy/paste control is desired, it can be managed at the developer level using named pasteboards. Likewise, the security risk of a user printing sensitive or classified data to the wrong printer can be managed when developing custom Apps.

Exploitable code errors

Most Apps contain software bugs, some of which can be exploited. Some bugs can be exploited in a way that allows code execution; others may cause the App to operate in a way other than it was intended.

Some common types of vulnerabilities that may be exploited in iOS Apps are:

- buffer overflows
- uncontrolled format strings
- use after free
- SQL injection.

Apple has implemented a number of anti-exploitation mitigations in iOS that make successful exploitation of the above vulnerabilities more difficult. However, it is still important for in-house App developers to understand the types of coding

errors that can lead to an exploitable vulnerability, and that operating system generic exploit mitigations do not provide complete protection for exploitable Apps.

Developers should refer to the following resources for more information:

iOS Security Starting Point

https://developer.apple.com/library/ios/referencelibrary/GettingStarted/GS_Security_iPhone/index.html

Apple Secure Coding Guide

https://developer.apple.com/library/ios/documentation/Security/Conceptual/SecureCodingGuide/

iOS Developer Library Security Topic

https://developer.apple.com/library/ios/navigation/#section=Frameworks&topic=Security

Mitigations

Though it is not possible to fully mitigate all the security risks mentioned, it is possible to substantially reduce their likelihood. This is done by:

- ensuring Apps that handle sensitive or classified data utilise appropriate iOS data protection classes
- configuring agency devices to disallow host pairing
- restricting Open-In behaviour between Managed Apps and Unmanaged Apps
- use of whole device Always On VPN or per-App VPN, in conjunction with appropriate firewall rules at the VPN end point
- ensuring that agency deployed Managed Apps are assessed for weaknesses before deployment.

Though the following guidelines specifically address Apps, they must equally be applied to App extensions.

Data Protection

If an App is required to write files containing PROTECTED data on a locked device from the background, such files should be protected using the Class B (*NSFileProtectionCompleteUnlessOpen)* data protection class. Otherwise all files that contain PROTECTED data, created or referenced by an App must be protected using the Class A (*NSFileProtectionComplete)* data protection class.

Keychain

If the App requires access to credentials on a locked device from the background, such credentials should be protected using the …AfterFirstUnlock or … AfterFirstUnlockThisDeviceOnly Keychain classes, otherwise credentials must be protected using the …WhenUnlocked or …WhenUnlockedThisDeviceOnly Keychain classes.

Questions to ask App developers

By asking the following questions, administrators put themselves in a good position to properly evaluate the security risks associated with installing a particular iOS App:

1. What is the flow of data throughout the App; source, storage, processing and transmission?
2. Which data protection classes are used to store data?
3. When data is transmitted or received, is it done through a secure means? Is Secure Transport used? If not why not?
4. What system or user credentials are being stored? Are they stored using the Keychain Class A? If not why not?
5. Are any URL Schemes or UTIs handled or declared?
6. Is the App compiled as a position independent executable (PIE)?
7. Is iCloud, or other cloud functionality used?

Note: For further information about iOS App assessment please refer to the iOS App Assessment Guide published on OnSecure (https://www.onsecure.gov.au). This guide is also available on request.

Managed Open-In

Despite best efforts to ensure that agency Apps protect data appropriately, having a single poorly designed App with registered document types can compromise the security of all the data on the device.

iOS features a pair of restrictions that help agencies control this security risk:

- "Allow documents from managed sources in unmanaged destinations"
- "Allow documents from unmanaged sources in managed destinations"

These restrictions can be used to control Open-In behaviour. In a deployment where:

- agency Apps are deployed by the MDM as Managed Apps
- classified files are only able to be accessed by the Managed Apps

Disabling the Managed to Unmanaged Open-In behaviour can be used to mitigate the security risk posed by users moving classified documents in to their own Unmanaged Apps.

Important: Agencies may allow unmanaged user App installation on PROTECTED iOS deployments if the *"Allow documents from managed sources in unmanaged destinations"* restriction is disabled.

Allowing unmanaged user App installation carries with it the following risks:

1. Improper utilisation of Unmanaged Apps for sensitive or classified work

   This sensitive or classified data may be recoverable if the data was not stored in an appropriate data protection class and the device was lost or stolen. There is also the possibility that such data may be transmitted over the network with or without transport security to an uncontrolled end point. This should be mitigated by use of Managed Apps, Managed Open-in restrictions and whitelisting of Apps.

2. Device exploitation via hostile App Store App

   An App Store App may have hidden functionality designed to gather and transmit personal or sensitive or classified data to a third party. It is also possible for an App Store App to maliciously execute code which is designed to exploit operating system vulnerabilities. Both security risks are significantly mitigated by the relatively hardened iOS runtime environment, whitelisting of Apps, and Apple's App Store review process.

Custom App development

Apps that are purpose built for agencies should use the following capabilities where possible:

- Use ProtectionComplete as a data protection class where possible. Data that needs to be synced in the background can

use ProtectionCompleteUnlessOpen, or ...AfterFirstUnlock, until the device is unlocked or the App is launched, at which point it can be moved into ProtectionComplete.

- The App delegate should register for and handle an “AppWillEnterBackground” event, so it can replace the current screen with a placeholder, ensuring that App opening animations do not leak sensitive or classified data

- The App should use NSURLSession where possible for communicating over the network to servers, and constrain its trust chain to the minimum viable set of CAs. The App should not degrade security from system defaults (TLS 1.2 with forward secrecy). This also enables the App to use SSO for network services. SSO credentials can be shared between multiple Apps, and ideally use identity certificates, rather than usernames and passwords.

- If deployed to TouchID enabled devices, the App should present an unlock screen, using an App password protected by TouchID. This secret should be stored in the Keychain using the local authentication framework, and a data protection class of ProtectionComplete with “DeviceOwnerAuthenticationWithBiometrics” set. See:

https://developer.apple.com/library/ios/samplecode/KeychainTouchID/Introduction/Intro.html

https://developer.apple.com/library/ios/documentation/LocalAuthentication/Reference/LocalAuthentication_Framework/

- The App should use a named Application pasteboard for copy and paste. Apps signed by the same developer identity can share named pasteboards. See:

https://developer.apple.com/library/ios/documentation/General/Conceptual/Devpedia-CocoaApp/Pasteboard.html

- Non-sensitive configuration information about an App should be configurable by MDM. See:

https://developer.apple.com/library/ios/samplecode/sc2279/Introduction/Intro.htm

CHAPTER 6

Suggested Policies

Our suggested policies for iOS devices used by Australian agencies.

Suggested Policies

This chapter lists suggested policies in graduated levels of response, applied to iOS devices at varying classifications. The agency's IT Security Advisor should be consulted for the specific usage scenarios for a deployment.

If iOS devices are being considered for use at classifications above PROTECTED, agencies must undertake a risk assessment following the guidance in the ISM as well as their own agency security policies and determine mitigation procedures and policy. Agencies must also obtain appropriate approval for any non-compliance in accordance with the ISM.

Feature	UNCLASSIFIED	UNCLASSIFIED WITH DLM	PROTECTED
Device selection	Agency's decision	A5 processor or later	A7 processor or later
BYOD (Bring Your Own Device)	Agency's decision	May be possible MDM opt-in for acceptable use policy agreement and enforcement recommended See ISM section on Mobile Devices	May be possible Supervision with Apple Configurator 2, MDM opt-in for acceptable use policy agreement and enforcement recommended See ISM section on Mobile Devices and BYOD
Device unlock with Passcode/TouchID	Must use passcode May use TouchID to unlock device	Must use passcode May use TouchID to unlock device	Must use passcode TouchID may be used by Apps once device is unlocked
Apple ID	May not be needed If iMessage/FaceTime is not used by agency, then Personal AppleID is sufficient	May not be needed If used can be personal or agency (if iMessage/FaceTime used)	May not be needed If used can be personal or agency (if iMessage/FaceTime used)
Home computer backup enforcement	Stated in agency usage policy	Stated in agency usage policy	Must use supervised mode and either prevent host pairing, or pair to a whitelist only

Table 6.1: Suggested policies

Feature	UNCLASSIFIED	UNCLASSIFIED WITH DLM	PROTECTED
iCloud	Refer to relevant ISM controls on Cloud services and data handling	Refer to relevant ISM controls on Cloud services and data handling iTunes Purchases and iTunes Match at agency discretion	Refer to relevant ISM controls on Cloud services and data handling No syncing documents and data by PROTECTED Apps No Backup of PROTECTED data No iCloud Keychain iTunes Purchases and iTunes Match at agency discretion
MDM	Optional depending on role of device/scale of deployment	Optional depending on role of device or scale of deployment Recommended for BYOD model	Recommended
Email	Third party email Apps which utilise native data protection and transport security may be used Recommend use of native Mail app	Third party email Apps which utilise iOS native data protection and transport security may be used Recommend use of native Mail app Exchange with certificate authentication	Third party email Apps must meet Chapter 5 data protection/keychain app requirements Recommend use of native mail app Must use Exchange with certificate authentication and should use TLS 1.2 for transport security
VPN-on-Demand	Optional depending on role per-App VPN preferred	Optional per-App VPN preferred	Required, if not using per-App VPN or Always On VPN

Table 6.1 (continued): Suggested policies

Feature	UNCLASSIFIED	UNCLASSIFIED WITH DLM	PROTECTED
Per-App VPN	Optional depending on role	Optional	Required, if not using Always On VPN
Always On VPN	Optional depending on role	Optional depending on role	Preferred
Secure browser	Optional	Optional	Recommended
User installation of App Store Apps	Agency's decision Recommend use of Managed Open-In	Agency's decision Recommend use of Managed Open-In	Possible Must use Managed Open-In restriction to prevent Managed to Unmanaged App document transfer
AirDrop	Agency's decision	Agency's decision	Should disable AirDrop.
AirPlay	AirPlay receiver should require passcode authentication May be managed by MDM.	AirPlay receiver should require passcode authentication May be managed by MDM.	AirPlay receiver should require passcode authentication May be managed by MDM.
Enterprise SSO	Agency's decision	Agency's decision	Recommend use of enterprise Apps which support this feature
Continuity	Agency's decision	Agencies need to assess the risks (i.e. sensitive information being transferred to a user's personal Mac)	Agency should disable Continuity through Configuration Profile Restriction

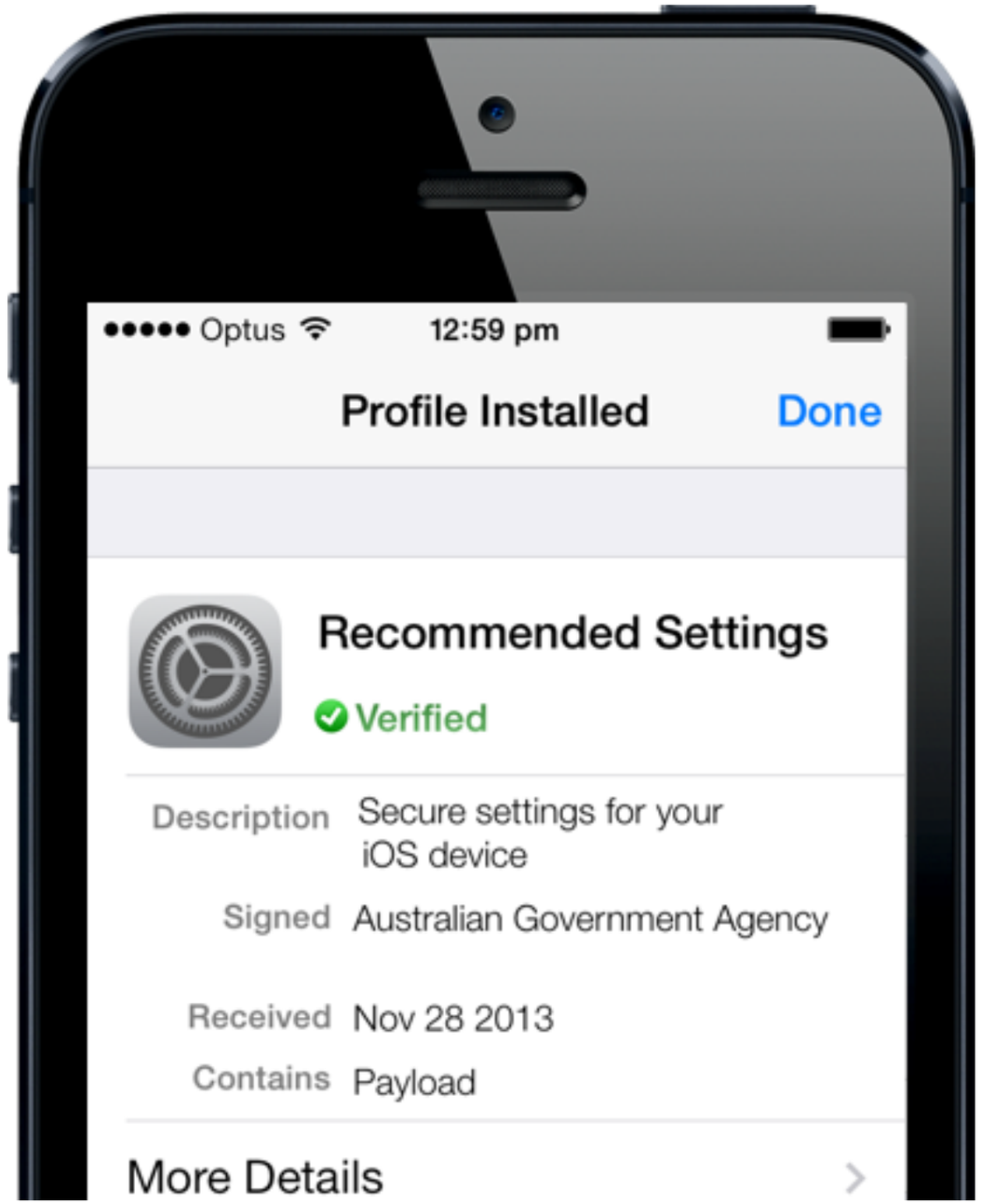
Table 6.1 (continued): Suggested policies

Feature	UNCLASSIFIED	UNCLASSIFIED WITH DLM	PROTECTED
Extensions	Agency's decision Recommend use of Managed Open-In	Agency's decision Recommend use of Managed Open-In	Possible Must use Managed Open-In restrictions to maintain managed/unmanaged container separation Extensions processing PROTECTED data are required to comply with the same guidelines as Apps
User storage of Home/ Health data on device	Stated in agency usage policy	Stated in agency usage policy	Recommend disallowing storage of personal Home/Health data, stated in agency usage policy
iCloud Drive	Agencies need to assess the security risks in their own environment	Possible under certain circumstances (i.e. files appropriately encrypted)	Disabled
Managed documents in iCloud	Agency’s decision	Possible Refer to relevant ISM controls on Cloud services and data handling.	Disabled
Touch-ID for in-house App authentication	Recommended in addition to existing passcode policy	Recommended in addition to existing passcode policy	Recommended in addition to existing passcode policy
Voice and SMS	UNCLASSIFIED only	UNCLASSIFIED only	UNCLASSIFIED only
SIM PIN	Recommended	Recommended	Recommended
Bluetooth	Disabled, unless required for specific business purpose.	Disabled, unless required for specific business purpose.	Disabled, unless required for specific business purpose.

Table 6.1 (continued): Suggested policies

CHAPTER 7

Recommended device settings



The profile settings that must be used when an iOS device is used on an Australian government network.

Recommended device settings

The settings described in this chapter are described in the context of using the Apple Configurator 2 graphical user interface to apply Configuration Profiles to a device. Apple Configurator 2 can not configure all profile types in its GUI - MDM implementations can often implement more. In addition, MDM management consoles may use different language to describe the setting from Apple Configurator 2.

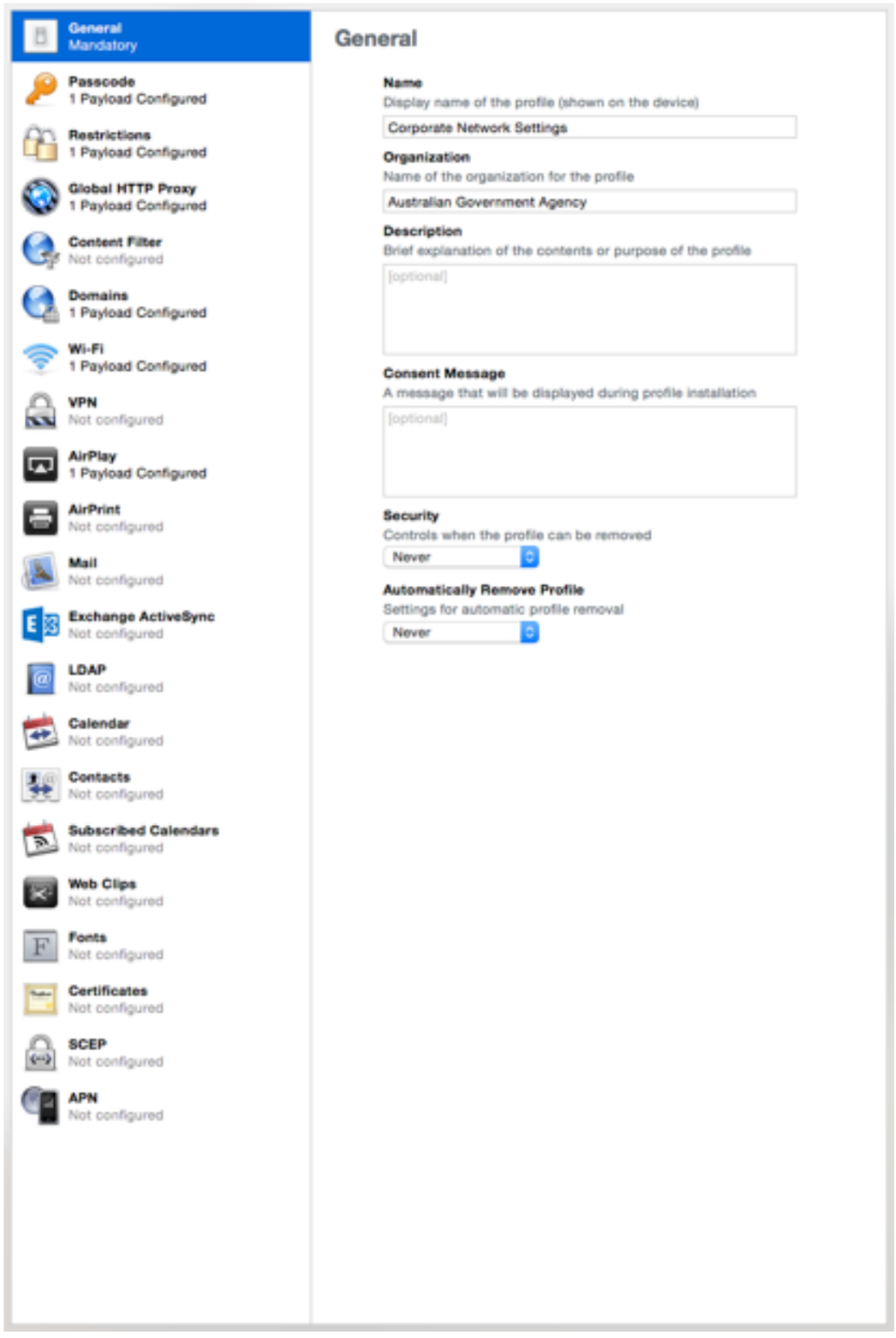
Note also that not all restrictions are present in the Restrictions Payload. For example, in order to properly configure Managed Open-in, where an agency and personal email account are both allowed on a device, but need to be separated, the following would need to be set in the Exchange Payload in addition to the Restrictions Payload. Deny “Allow messages to be moved” as well as “Allow Mail Drop”. Also use certificate based authentication rather than username/password, *AND* use TLS and/or S/MIME.

Another example would be in an IKEv2 VPN payload, where the local exception list for processes allowed to send traffic outside the VPN tunnel is enumerated in the VPN payload, not in the Restrictions Payload.

The restrictions listed are suitable for agency owned devices carrying data at PROTECTED.

Restrictions that are supervised only can not be applied to BYOD devices, and are not required at UNCLASSIFIED with DLM.

General Settings



Setting	Recommendation
Security	• Profile Security should be set to “Always” if setting is for the convenience of users accessing non-sensitive data (e.g. a subscribed calendar of Australian public holidays). Opt-In MDM profiles would usually fit into this category as well. • Profile security should be set to “With Authorisation” for profiles that IT staff can remove temporarily. Generally users would not receive the passcode to such profiles. • Most profiles that are not MDM managed should be set to “Never”. The passcode policy profile, if used, should be set to “Never”.
Automatically Remove Profile	This is not required when MDM is used as the exclusive mechanism to deploy/remove profiles. Configuration Profiles can be set for automatic removal on a specific date or after a defined interval. This function can be used at agency discretion, typical use cases include: • Guest user profile • Temporary profile for overseas travel

Passcode

(Can be set via EAS or Configuration Profile)

Setting	Recommendation
Allow simple value	Disabled
Require alphanumeric value	Enabled
Minimum passcode length	8
Minimum number of complex characters	0
Maximum passcode age	90 days
Auto-lock	5 minutes
Passcode history	8
Grace period for device lock	None
Maximum number of failed attempts	8*

* Passcode exponential back off timing begins after 5 attempts. Allowing 8 attempts will require users to wait 21 minutes cumulatively before forcing a device wipe.

Note: Depending on the EAS version, only some of the above may be set by the EAS Server and a Configuration Profile would be required.

Restrictions (Functionality)

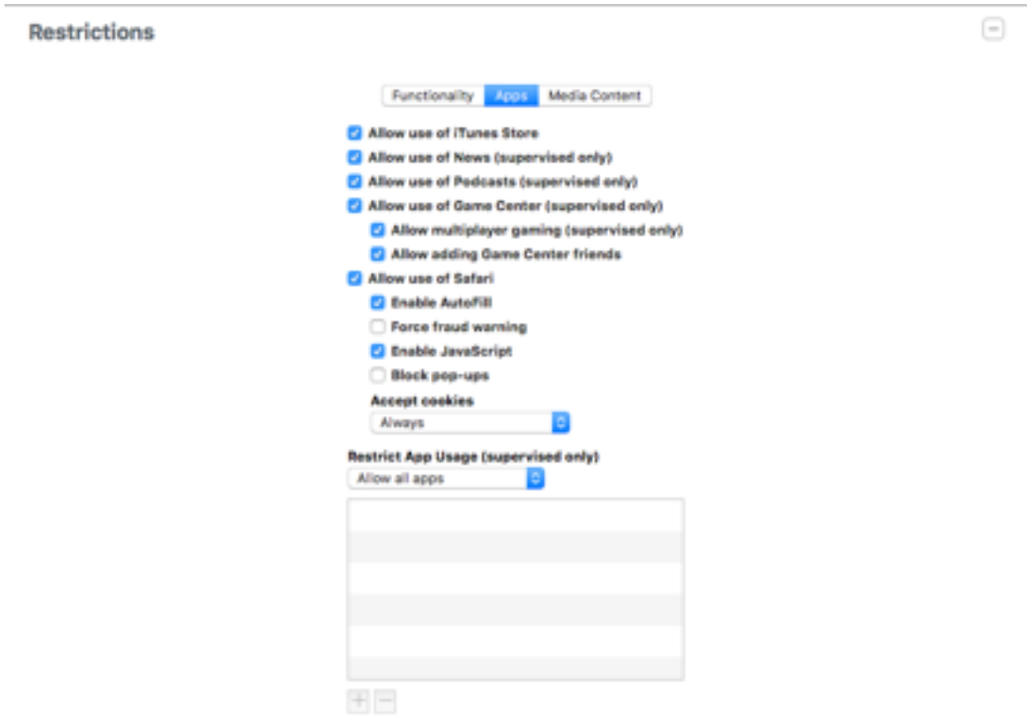
Setting	Recommendation
Allow use of camera	Up to agency, depending upon acceptable use policy.
Allow FaceTime	Up to agency. AppleID should be institutional if not UNOFFICIAL. Communication must be Unclassified.
Allow screenshots	Disabled
Allow Airdrop	Disabled
Allow iMessage	Up to agency. AppleID should be institutional if not UNOFFICIAL. Communication must be Unclassified.
Allow voice dialing	Up to agency
Allow iBooks Store	Up to agency. Depending upon acceptable use policy.
Allow installing Apps	Up to agency. May be enabled at PROTECTED when used with “Managed Open-In” restrictions described below.

Setting	Recommendation
Allow removing Apps	Up to agency. Disabling may be used to prevent users from removing business critical Apps.
Allow In-app purchase	Up to agency. Depending upon acceptable use policy.
Require iTunes Store password for all purchases	Enabled
Allow iCloud backup	Disabled. Note that this can be disabled per-app for Managed Apps as an alternative.
Allow iCloud documents and data	Disabled. Note that this can be disabled per-app for Managed Apps as an alternative.
Allow Managed Apps to store data in iCloud	Disabled unless Managed Apps meet ISM requirements for reducing storage/handling requirements of the remotely stored data to Unclassified.
Allow iCloud keychain	Disabled. Note that this can be disabled per-app for Managed Apps as an alternative.
Allow backup of enterprise books	Up to agency. Recommend disabled when agency distributes enterprise iBooks containing sensitive data.
Allow notes and highlights sync for enterprise books	Up to agency. Recommend disabled when agency distributes enterprise iBooks containing sensitive data.

Setting	Recommendation
Allow iCloud photo sharing	Up to agency. If enabled, photos taken with the built in camera App may be backed up to Apple's iCloud infrastructure and distributed to another user's iOS devices.
Allow My Photo Stream	Up to agency. If enabled, photos taken with the built in camera App may be backed up to Apple's iCloud infrastructure and distributed to a user's other iOS devices.
Allow automatic sync while roaming	Enabled
Force encrypted backups	Enabled
Force limited ad tracking	Enabled
Allow Erase All Content and Settings	Disabled unless required for users with a specific need to erase devices themselves.
Allow users to accept untrusted TLS certificates	Disabled
Allow automatic updates to certificate trust settings	Enabled
Allow configuring restrictions	Disabled
Allow installing Configuration Profiles	Disabled unless required for agency deployment

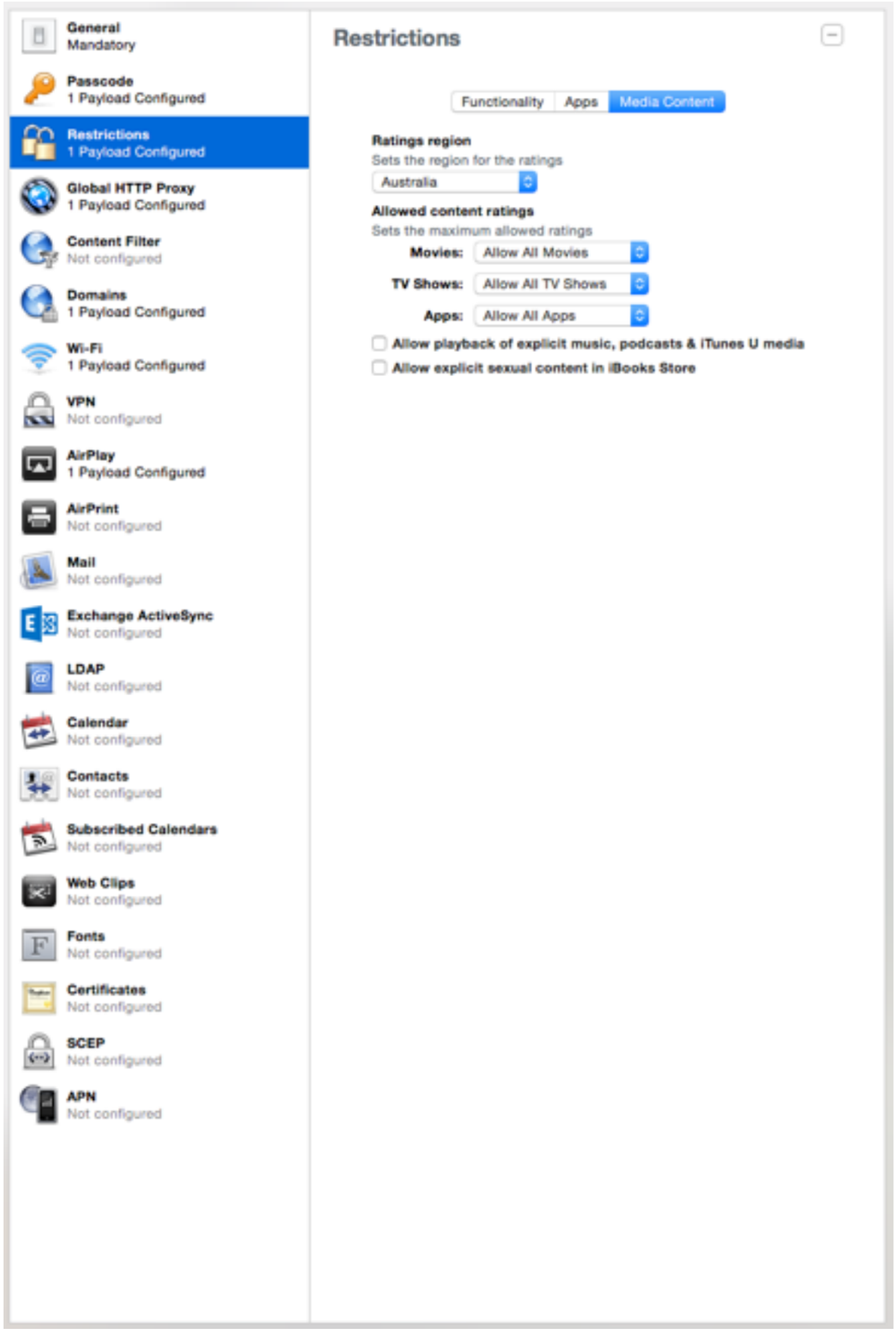
Setting	Recommendation
Allow modifying account settings	Enabled
Allow modifying cellular data app settings	Up to agency. Note B2B and Enterprise Apps can do this explicitly at the code level.
Allow modifying Find my Friends settings	Enabled
Allow pairing with non-Configurator hosts	Disabled
Allow documents from managed source in unmanaged destinations	Disabled. This control helps to prevent documents being opened in unmanaged user applications.
Allow documents from unmanaged sources in managed destinations	Up to agency. If enabled, a user may open documents from their own unmanaged Apps into agency managed Apps. If enabled, there is a risk that a managed app could be exploited by hostile content from an unmanaged user application. Recommended disabled at PROTECTED.
Allow Handoff	Disabled
Allow Internet results in Spotlight	Up to agency. Recommended disabled.
Send diagnostic and usage data to Apple	Disabled

Setting	Recommendation
Allow Touch ID to unlock device	Disabled at PROTECTED. TouchID can be used to unlock Apps after device is unlocked with a passcode.
Require passcode on first AirPlay pairing	Enabled
Allow Passbook notifications while locked	Up to agency. If enabled, passbook items may be accessible from the lock screen.
Show Control Center in lock screen	Up to agency. If enabled, Wi-Fi and Bluetooth may be enabled or disabled from the lock screen.
Show Notification Center in lock screen	Disabled
Show Today view in lock screen	Disabled
Allow Siri	Up to agency. All uses of Siri and Siri dictation must be treated as Unclassified.
Allow Siri while device is locked	Disabled
Enable Siri profanity filter	Up to agency. Depending upon acceptable use policy.
Show user-generated content in Siri	Up to agency. All uses of Siri and Siri dictation must be treated as Unclassified.



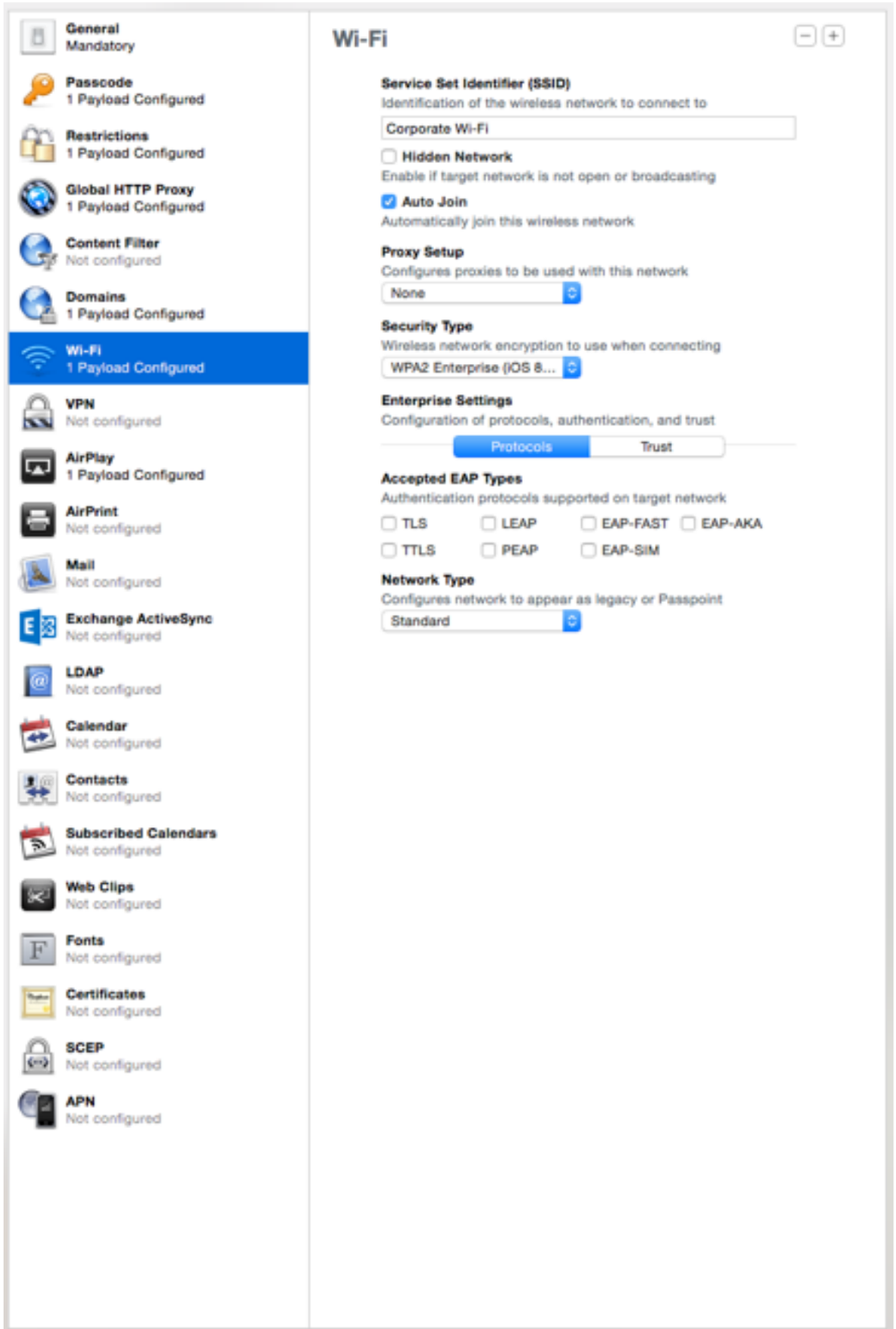
Restrictions (Applications)

Setting	Recommendation
Allow use of News	Up to agency. Depending upon acceptable use policy.
Allow use of iTunes Store	Up to agency. Depending upon acceptable use policy.
Allow use of Podcasts	Up to agency. Depending upon acceptable use policy.
Allow use of Game Center	Up to agency. Depending upon acceptable use policy.
Allow multiplayer gaming	Up to agency. Depending upon acceptable use policy.
Allow adding Game Center friends	Up to agency. Depending upon acceptable use policy.
Allow use of Safari	Enabled
Enable AutoFill	Enabled
Force fraud warning	Enabled
Enable JavaScript	Enabled
Block pop-ups	Enabled
Accept cookies	Up to agency. Recommend “From current website only”
Restrict App Usage	Up to Agency based on approach taken to App whitelisting.



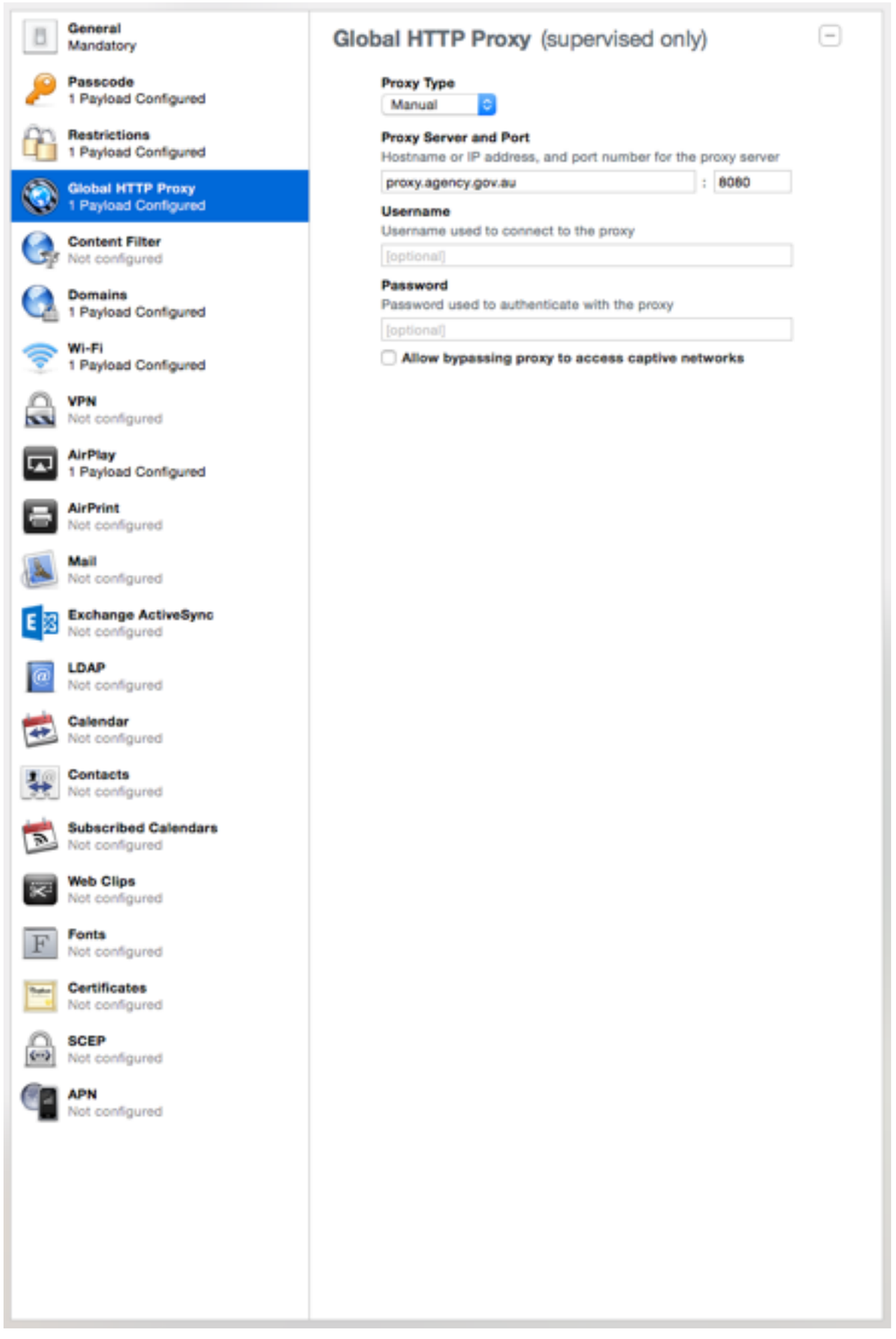
Restrictions (Media Content)

Setting	Recommendation
Ratings Region	Australia
Allowed content ratings (all)	Up to agency. Depending upon acceptable use policy.
Allowed content ratings (all)	Up to agency. Depending upon acceptable use policy.
Allow explicit music, podcasts & iTunes U	Up to agency. Depending upon acceptable use policy.
Allow explicit sexual content in iBooks Store	Up to agency. Depending upon acceptable use policy.



Wi-Fi

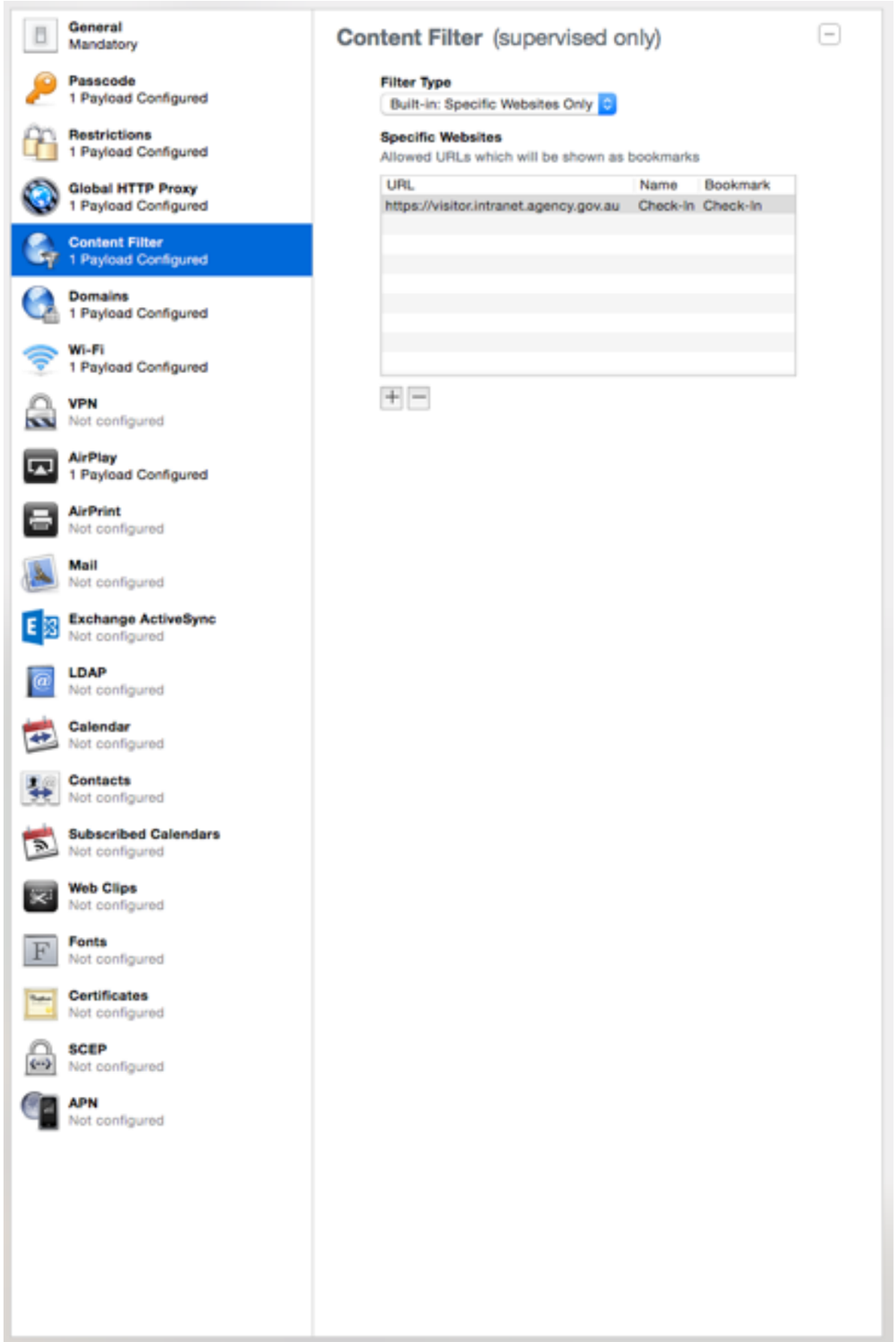
Setting	Recommendation
Service Set Identifier (SSID)	As appropriate for agency network
Hidden Network	As appropriate for agency network
Auto Join	Recommended enabled
Proxy Setup	As appropriate for agency network
Security Type	WPA2 Authentication with EAP-TLS and a pre-shared key as a minimum, per use RADIUS or 802.1X recommended.
Enterprise Settings	Protocols, authentication and trust to match network requirements. 802.1X with device identity certificate and username/ password is the preferred authentication mechanism for Unclassified (DLM) and higher.
Network Type	Standard



Global Proxy

Setting	Recommendation
Proxy Type, Server, Port, Username and Password	As appropriate for agency network. Consider use of AOVPN with transparent proxy as an alternative to Global Proxy
Allow bypassing proxy to access captive networks	Up to agency. If enabled, the proxy setting will be bypassed when an iOS device assesses the connected network to be “captive” (e.g. Paid hotel or airport Wi-Fi). For most usage scenarios enabling this defeats the purpose of using global proxy. Recommended disabled.

Global Proxy only impacts a subset of Apps that call specific web services APIs, and is really short hand for setting a proxy separately in VPN, Wi-Fi and Cellular payloads. Use of a content filter or VPN and a transparent proxy is a more reliable way of capturing traffic.



Content Filter

The content filter configuration payload has 3 different types, which may be used to :

- limit access to adult websites via a whitelist or blacklist
- to restrict web access to a whitelist of specific allowed websites
- route all traffic through a 3rd party filter prior to transport layer encryption being applied.

These restrictions are applied universally regardless of which network interface is used. Use of these restrictions is up to the Agency.

Domains

The Domains payload performs three roles:

- marking external email addresses in Mail app
- ensuring that Safari treats downloaded documents as “Managed” while browsing managed web domains
- controlling which domains Safari is allowed to autofill passwords for (note that this is a whitelist not a blacklist)

Managed Email Domains consists of a whitelist of string matching expressions for email suffixes. When this payload is configured, the iOS Mail App will highlight email addresses which are not in the whitelist.

Managed Web Domains consists of a list of string matching expressions for URLs.

A good example of how this may be used with intranet domains might be to add “*.intranet.agency.gov.au” which would match:

- https://intranet.agency.gov.au/selfservice
- https://wiki.intranet.agency.gov.au

It would not match the internet facing agency website:

- https://www.agency.gov.au

Note: It is strongly recommended that Managed Web Domains be configured to lower the likelihood of sensitive data breach.

VPN

IPSec and TLS are Approved Cryptographic Protocols, please refer to the ISM for more information:

http://www.asd.gov.au/infosec/ism/

To help determine the server side settings that iOS supports, refer to the iOS Deployment Technical reference at:

http://www.apple.com/iphone/business/it/deployment.html

Prior to iOS 7, ASD's recommended On Demand configuration was to trigger on a URL whitelist using the *OnDemandMatchDomainAlways* Configuration Profile rule. In iOS 7 this rule has been deprecated and replaced with the new *EvaluateConnection* rule set. To create a VPN profile that works on both iOS 7 and earlier releases, it is necessary to utilise both *EvaluateConnection* and *OnDemandMatchDomainAlways* together. In such a configuration, iOS 7 and later will use *EvaluateConnection* while previous releases will ignore it.

There are several VPN On-Demand configurations that are possible at PROTECTED. Examples are:

- Action:Connect on DNSDomainMatch:(array of whitelisted domains OR wildcard)
- Action:Connect on InterfaceTypeMatch:Cellular, Action:EvaluateConnection on InterfaceTypeMatch:Wi-Fi Domains:(whitelist or wildcard) DomainAction:ConnectIfNeeded RequiredURLStringProbe: (trusted internal HTTPS server)

Following the release of iOS 9, ASD recommends use of Always on VPN or Per-App VPN in preference to On Demand VPN.

Always-on VPN

The preferred VPN configuration for devices containing PROTECTED data is IKEv2 IPSsec Always-On VPN. It is available as a supervised only option for IPSec IKEv2 VPN. It ensures that all network traffic is routed to the VPN. When the VPN can't be established, no network traffic is be transmitted. Note that AOVPN will bock access to carrier services like MMS and voicemail, unless explicitly whitelisted.

Always-On VPN should be used in PROTECTED deployments, and is the simplest and most secure VPN configuration solution at this classification.

Always on VPN is the Common Criteria evaluated configuration, and fails closed (i.e. if the VPN can not be established, the device acts if there is no network connection)

Per-App VPN

Per-App VPN is most commonly used where a device has a mixture of personal and institutional content. Apps containing work related data are installed by MDM as managed Apps,and configured to use per-App VPN. Use of the built in IPSec IKEv2 VPN client for per-App VPN is recommended for PROTECTED content, and it is the simplest to configure device side, as there is no dependency on installing a VPN client App. Third party VPN agent Apps that support per-App VPN and use AACP and AACA can be used at UNCLASSIFIED with DLM. Per-App VPN usually can't be used concurrently with other VPN types.

Per-App VPN fails closed for the Apps it has been applied to by policy. Note that per-App VPN can also be applied to individual tabs in Safari connecting to specified domains or URLs.

AirPlay Mirroring

The AirPlay mirroring payload may be used to pre-populate a device with a whitelist of AirPlay devices and AirPlay passwords. These options can be configured to assist in controlling access to AirPlay resources. For AirPlay to AppleTV, it is recommended that agencies utilise alphanumeric passcodes to prevent unauthorised use of AirPlay.

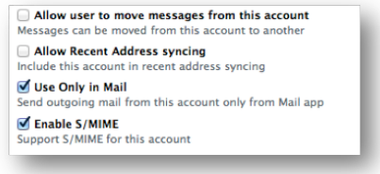
Reminder: AirPlay should never be used to transmit video/audio of a classification greater than that of the network the device is using.

AirPrint

The AirPrint payload may be used to pre-populate a device with a list of discoverable or undiscoverable AirPrint devices. These settings are intended for user convenience.

Mail

A Mail payload is not typically needed if EAS (e.g. Exchange ActiveSync Gateway, Lotus Notes Traveller) is in use. The Mail payload can co-exist with Exchange. If used, fill with settings appropriate to agency network.



Setting	Recommendation
Use SSL	Enabled, with authentication.
Allow user to move messages from this account	Disabled in deployments where multiple email accounts of different classifications are expected.
Allow Recent Address syncing	Disabled
Use Only in Mail	Enabling this setting prevents other Apps from sending mail with this account. Though enabling this option is preferable, it will break functionality in some third party Apps – including some MDM client Apps. Enable this setting if possible in your deployment.
Enable S/MIME	Enabled if agency infrastructure supports S/MIME

Exchange ActiveSync

The EAS server settings should be filled in as required for the agency network noting the following consideration:

- Authentication credentials required to control which device and which users have access to EAS must be added to the Certificate/Credentials Payload. The credential required for authenticating the ActiveSync account can then be selected.

Note: If a profile with an EAS payload is removed, all EAS synced email and attachments are deleted from the device.

Setting	Recommendation
Use SSL	Enabled, with authentication.
Allow user to move messages from this account	Disabled in deployments where multiple email accounts of different classifications are expected.
Allow Recent Address syncing	Disabled
Use Only in Mail	Enabling this setting prevents other Apps from sending mail with this account. Though enabling this option is preferable, it will break functionality in some third party Apps – including some MDM client Apps. Enable this setting if possible in your deployment.
Enable S/MIME	Enabled if agency infrastructure supports S/MIME

LDAP

LDAP settings should be filled in as required for the agency network. LDAP is not typically needed if Exchange Global Address List (GAL) is used, but can co-exist.

- The “SSL Enabled” option toggles support for TLS. This should be enabled if supported by agency infrastructure.

Calendar (CalDAV)

CalDAV settings should be filled in as required for the agency network. CalDAV may not be needed if Exchange is used, but can co-exist.

- TLS Enabled if supported by agency infrastructure.

Contacts (CardDAV)

CardDAV settings should be filled in as required for the agency network. CardDAV may not be needed if Exchange is used, but can co-exist.

- TLS Enabled if supported by agency infrastructure.

Subscribed Calendars

Subscribed Calendars settings should be filled in as required for the agency network.

- TLS Enabled if supported by agency infrastructure.

Web Clips

Web Clips are “aliases” or links to URLs with a custom icon that can be installed on a device home screen. Settings should be filled in according to the agency’s deployment requirements.

- Typical use would include links to pages for AUP, helpdesk contact details, telephone URLs, and SCEP re-enrolment pages. Note that these web pages could use preference manifest settings in their HTML to work when the site is offline or the device is off the network.
- Web clips can also be used to install Enterprise In-House Applications.

Certificate (Credentials)

Include SSL chain of trust back to the root CA certificate, including intermediates.

SCEP

Simple Certificate Enrolment Protocol (SCEP) is normally configured during Over-The-Air MDM enrolment, however the Configuration Profile SCEP payload can be used when pre-configuring SCEP enrolment prior to device issue. Configure as necessary for agency deployment.

Advanced/APN

If an APN is used, settings should be filled in as required for agency network noting the following considerations:

- Access Point User Name and Password must be used.
- Proxy should be configured as appropriate.

Details should be discussed with your telecommunications carrier.

CHAPTER 8

Mobile Device Management

How to use Mobile Device Management software to provision, configure and manage iOS devices.

Mobile Device Management

Organisations should use Mobile Device Management (MDM) software to provision, configure and manage iOS devices. Basic MDM functionality can be obtained using OS X Server Profile Manager, however it does not scale to enterprise deployments. Profile Manager is best used in a development or test environment for testing pre-release builds of future versions of iOS. Third party MDM software operated at much greater scale, and often features extended functionality.

Functions

There are five categories of tasks which are implemented under the Apple MDM protocol:

- device enrollment
- device configuration
- App configuration
- device query
- device command.

Most third party MDM servers implement the functions which are supported natively by iOS, some extend the feature set with a client app. This chapter describes the useful functions which should be common to most MDM implementations.

Workflow Overview

Before a device can be managed it must be enrolled with an MDM server. There are a variety of ways that this can be performed

- Device Enrollment Program
- MDM agent App
- Web Portal.

If devices are institutionally owned, then organisations should use Device Enrollment Program. The device will be enrolled in MDM whilst still in the setup assistant. The most common recommended configuration is that the device be supervised over the air, MDM enrollment be mandatory, and MDM enrollment be non-removable. The MDM enrollment can be generic (the same for all devices) or personalised (customised for a specific user's requirements e.g email configuration, identity certificates) Note that Apple Configurator can be used in conjunction with DEP, and this is most commonly done for shared devices that are not personalised to a specific user.

If a device is user owned, then the user should download the MDM agent App from the iOS App Store, and enroll the device. In this workflow MDM will be user-removable.

Deployment with Supervision

In conjunction with Device Enrollment Program, MDM enrollment and supervision can occur over the air, without returning the device to an Apple Configurator workstation and connecting via USB. However, if the devices used are not part of Device Enrollment Program, they should be supervised with Apple Configurator. For PROTECTED deployments, devices MUST be put in to supervised mode, it is recommended that all devices be put in supervised mode, as this permits MDM to manage Activation Lock.

User owned devices would not normally be supervised, as this involves erasing the device, and enabling invasive policy to be applied. This means that user-owned devices would not normally be allowed to connect to a PROTECTED network or contain PROTECTED data, and be limited to UNCLASSIFIED with Dissemination Limiting Markers such as FOUO. For user owned devices to carry PROTECTED information, they would need to be prepared for deployment in a similar way to non DEP institutionally owned devices (discussed further in the section on device sanitisation).

Managing iOS Devices

After devices have been deployed, administrators may need to change or remove Configuration Profiles. Apple's MDM protocol allows MDM servers to remove and install profiles, but only under the following conditions:

1. The MDM Payload must have the "Allow installation and removal of Configuration Profiles" access right set.
2. The Configuration Profile in question must have been installed by the MDM.

In some cases devices may simply need to be reconfigured to support a change in infrastructure or to update certificates, however this functionality can also be used to add or remove more restrictive settings on demand (eg to temporarily allow host pairing). It should be noted that some third party MDM vendors provide a geo-fencing function, which installs or removes Configuration Profiles depending upon location or other triggers (eg integration with a access control system of a building).

If a device is lost or stolen, an MDM administrator can take several actions to protect agency information. One option available is to issue a remote wipe command. This action replaces the file system key with a new randomly generated key, permanently rendering all data in the file system irrecoverable. If Find My iPhone is enabled, then the device is effectively a brick, with value only for spare parts. Note also that if wiped, then the agency has no capacity to monitor the status or location of the device from that point.

If a BYOD device is stolen, an MDM administrator can either wipe the device as above, or they can un-enroll the device from MDM, or if the MDM supports it, perform an "Enterprise Wipe". In both cases, all enterprise Apps & their data, and accounts (including email) and their data, are deleted from the device , leaving only personal content. The difference between these approaches is that a partial wipe may leave some data (particularly Class D , and potentially Class C data) that is forensically recoverable, if control of the device is lost, and the

passcode of the device is known to the possessor, and the device is not rebooted prior to forensic acquisition attempts.

In the event that a user forgets their passcode, the Apple MDM protocol allows for a remote clear passcode command. Not only does this command unlock the device, by removing the passcode it also disables data protection. Consequently, the issuing of this command immediately modifies the storage and handling requirements of the device to that of the maximum classification of data stored on the device.

Note: The clear passcode command must not be issued unless the device owner is in physical possession of the device. The clear passcode command must never be issued to a lost or stolen device.

There are a number of methods that are normally used for de-provisioning iOS devices. These may involve remotely removing MDM and associated profiles or issuing a remote wipe. In PROTECTED deployments, iOS devices should be returned to base and de-provisioned using the methods described in Chapter 4.

Querying Devices

There are a number of query operations supported by Apple's MDM protocol, which can be used to ensure that devices remain in compliance with agency policy. As an example, it is possible to query a device to find out:

- the device's iOS version
- which Configuration Profiles are installed
- the presence/complexity of passcode
- which Apps are installed.

Some MDM implementations may allow for predefined or scripted actions to take place when a device is found to be out of compliance with policy.

Managed Settings

Though iOS device configuration is almost entirely managed using Configuration Profiles, it is possible for an MDM server to modify certain specific options in appropriately written Apps without Configuration Profiles. These managed settings may be modified at any time and without user interaction. Managed settings may only be used if the "Apply Settings" right is set in the MDM payload.

Managed Apps

An MDM server may issue an Install Application command via the Apple MDM protocol. This command contains either:

- an app's iTunes store ID for App Store Apps
- a URL link to an App's manifest XML file for in-house or custom Apps.

Additionally, the command must specify how the app is to be managed. Namely, whether the app is to be removed when the MDM profile is removed and whether app data can be backed up. After receiving a valid Install Application command, an iOS

device will prompt a user to accept the app installation. Apps that are installed in this way are called Managed Apps.

When Apps require payment, it is possible for an MDM to provide a VPP redemption code, a Managed Distribution Licence code, or assign an App to a device. In many third party MDM implementations this function is tightly integrated with an enterprise app store. iOS 9 permits MDM to install App Store Apps while still blocking the user from installing them.

In iOS 9, MDM can also "take over" user installed Apps and make them managed, if the MDM has a licence for the App. This feature should be used with caution.

Finally, it is possible for an MDM server to issue a Remove Application command to remotely remove a managed app and its data. Apps that were not installed as a managed app by the MDM cannot be removed in this way.

Choosing MDM Software

At the time of writing this guide there are over 100 vendors shipping MDM solutions that have at least some support for Apple's MDM protocol. Some of these MDM solutions focus on the core functionality provided by Apple's MDM protocol, others enhance this by providing additional features via a client app. Many MDM vendors distribute an MDM solution that can manage multiple mobile and desktop client platforms. The following information is provided to help agencies understand functionality commonly offered by vendors and to describe the advantages and risks of various deployment options.

In line with the ASD Top 4 *Strategies to Mitigate Targeted Cyber Intrusions*, careful consideration should be given to choice of MDM vendor, as vendor choice can determine an agency's ability to comply with the Top 4. One of the critical assessment criteria in MDM vendor choice is aggressive support for new versions of the platform being managed - this permits client devices to be updated to the latest iOS patch level in a timely way, as patches become available, rather than deferring updates for months until the MDM vendor catches up. In addition, it means the vendor will be supporting the controls associated with new features in the platform. An MDM vendor whose release strategy delays patching the end point devices by many months, or in some cases years, is generally unsuitable for government use-cases. This is most common among MDM vendors who licence MDM for free, due to the customer buying some other licence or service from the vendor.

Secondly, agencies should consider the capability of the MDM server, their infrastructure, and their App strategy to force rapid iOS and App updates. Operating devices purely off cellular, without use of Wi-Fi can limit or delay the deployment of updates (due to budgetary constraints on cellular data usage, and that devices normally only automatically download updates when connected to Wi-Fi). Use of SDKs to manage Apps must be considered carefully as this can delay updating Apps (due to size, and scheduling development effort). Wi-Fi infrastructure should be configured to allow devices to upgrade iOS and Apps. Finally MDMs that support the MDM protocol commands to force a download of a new iOS version, and force the update of a Device assigned (rather than AppleID assigned) VPP app, should be considered strongly.

Proprietary Functionality

MDM software vendors may have a client app that can interact with an MDM server. Such Apps can interact with a device in ways beyond that which the Apple MDM protocol allows for. These MDM client Apps do not operate at any elevated level of privilege, and if installed from the App Store are subject to normal App Store approval processes.

Email attachment security is a feature that some MDM vendors provide outside of Apple's MDM protocol. The purpose of this function is to prevent the iOS Mail app "Open In" functionality from opening a sensitive attachment in an untrusted app. One way that this type of service can work is by an email proxy transparently encrypting attachments and then having the file opened by a client app that has registered the appropriate file or data type information with iOS. Typically this allows the file to be opened and decrypted by the client app from the iOS Mail app using "Open In" functionality. Managed Open-In can be used to achieve a similar outcome.

A valuable proprietary feature that some MDM vendors provide is an Enterprise App Store. This is a web portal, that allows a user to pull the whitelisted Apps that they require rather than have Apps pushed at them. An Enterprise App Store may exist as a mobile web application or a native app initially pushed to a device. In both cases this functionality requires use of native iOS MDM protocol functionality, but may contain proprietary extensions. Such Enterprise App Stores often allow administrators to distribute not only in-house applications but also App Store Apps and Volume Purchase Program (VPP) Apps.

Frequently MDM vendors also provide a secure container client app. The purpose of this type of app is to act as a secure repository for common file types. Most implementations allow users to view common file types while some allow file editing. The way in which these types of Apps provide security for files varies and can include:

- standard iOS data protection
- encryption provided by iOS cryptography libraries
- encryption provided by 3rd party cryptographic software.

If the container solution uses 3rd party cryptographic software, and does not use iOS data protection, then should be subject to a ACE if intended to be used for PROTECTED data.

Important: Class A data protection in iOS is sufficient for PROTECTED data.

On-premise and MDM Software as a Service

In addition to on-premise options, several MDM vendors offer as a Software as a Service (SaaS) Cloud offering. The benefit of this solution is that it frees administrators from having to set up and maintain a MDM server and often may offer a lower cost of ownership. In some cases a SaaS MDM solution may have a tightly integrated client app that improves overall user experience and reduces administrative burden.

Although there are many advantages to this MDM model, there are also significant risks. At a fundamental level, an MDM server controls access to agency information and authority over the configuration of the devices it manages. For example, an MDM routinely processes or stores:

- key material that can be used to unlock a device that is passcode protected
- Configuration Profile data for MDM enrollment and credentials for access to an agency's network.

Since key material is classified at the same level of the data it is protecting, the systems used by the MDM service provider must be hardened and accredited to the same minimum standard as the sponsoring agency's systems.

Should the MDM be compromised, an adversary may be able to perform all tasks an MDM is capable of. This may include:

- using escrow keybag material to unlock passcode protected devices
- provisioning unauthorised devices
- distributing malicious Apps.
- clearing passcodes from specific devices

An adversary may also be able to leverage information in Configuration Profiles to establish VPN or Wi-Fi connections to agency networks.

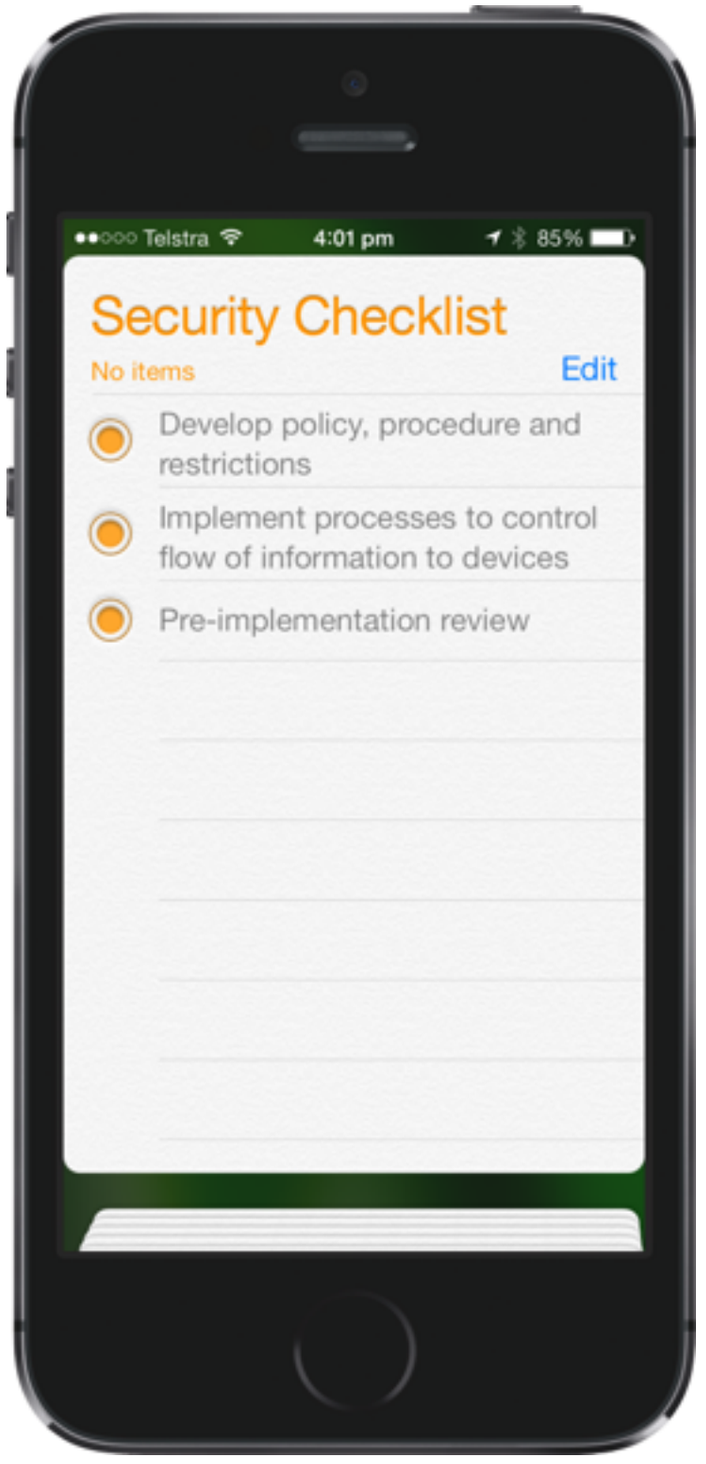
To protect against the risk of unauthorised access by a third party, administrators are also strongly advised to read ASD's guidance on Cloud computing security when considering a SaaS MDM deployment. Agencies with devices handling UNCLASSIFIED with DLM and PROTECTED data must ensure they are compliant with the controls in the ISM that address Cloud security, in particular "Outsourced Cloud Services" and "Outsourced General Information Technology Services". For security guidance on cloud services, the list of existing Cloud services certified by ASD, and the ISM, see

http://www.asd.gov.au/infosec/cloudsecurity.htm

http://www.asd.gov.au/infosec/ism/

CHAPTER 9

Security Checklist



Ensure that all key tasks in securely deploying iOS devices have been completed.

Before Deploying iOS Devices

Develop agency policy and procedures, including any restrictions, for the use of iOS devices that align with Australian Government legislation, policies and standards, and that adhere to Australian government requirements.

Effective policies and procedures help to ensure that an agency considers relevant issues and operates in accordance with legislation and whole-of-government guidelines. Documenting and making these available to users will help ensure that users are aware of an agency's expectations of them when using mobile devices. On iOS devices, placing a policy Web Clip on the device makes it highly accessible to the user.

Implement processes to security classify, protectively mark, and control the flow of information that may be transmitted to/from the iOS device.

Email filtering solutions can filter and mark email based on header metadata and shorthand notation in the subject line. Agencies must security classify and protectively mark all email, and controls must be implemented at email servers and gateways to restrict delivery of inappropriately classified information to and from an agency, including to mobile devices.

Undertake an iOS device pre-implementation review.

Agencies deploying iOS devices may consider undertaking a pre-implementation review. This review would assess the planned deployment strategy, mitigation controls, policies and procedures against the requirements defined in the relevant policy and guidance documents.

Manage Use of iOS Devices

Provide users with training on the use of iOS devices and security requirements.

In many areas of administration, failure to follow policies and procedures is not a result of deliberate actions, but a lack of awareness of requirements. Training can assist users to implement policies and procedures. The existence of training can also help distinguish deliberate misuse from incompetent usage. As part of this training agencies should also inform users that these devices are likely to be an attractive target for thieves, and that the implications of the information contained in them being accessed by others could be detrimental to the Australian government.

Ensure that users formally acknowledge their agreement to adhere to agency specific Acceptable Usage Policy and procedures.

Users must be aware of and agree with the agency's policy and procedures. The ramifications of failing to apply those policies and procedures must also be clear to users.

Ensure that users classify and protectively mark all email with the highest classification of the content or attachment, in accordance with Australian government standards.

Users must be conscious of the security classification of information that they are sending to or from mobile devices. Agencies must ensure that users classify and protectively mark all agency-originated email or attachments in accordance with the highest classification of the content.

Infrastructure Considerations

Server infrastructure for EAS, MDM, Web and associated CA infrastructure that supports an iOS deployment must be controlled, either directly or under contract, by the Australian government and appropriate for the classification the network operates at.

These servers should be situated in a controlled environment, and will permit the implementation of consistent policy and device settings.

In many cases, SaaS solutions may not be acceptable for iOS MDM deployments.

Agencies must ensure that content is transferred between an iOS device and an agency's ICT systems in accordance with Australian government policy.

Email protective marking filtering mechanisms must be implemented to provide a higher level of security by automatically preventing information of an inappropriate classification being sent to a mobile device. These mechanisms are described in the *Email Protective Marking Standard Implementation Guide for the Australian Government*, available at:

http://www.finance.gov.au/files/2012/04/email_pmsig.pdf

Ensure that email originating outside the agency is not sent to the iOS device unless it is classified and labelled appropriately.

Communications originating outside the agency may also include classified information. The policies and standards applied to external communications must also be applied to internally generated information. Emails that do not have protective markings should not be transmitted to mobile devices. Agency policy may define a subset, e.g. an agency may only permit Unclassified information to be forwarded to a mobile device. These mechanisms are described in the *Email Protective Marking Standard Implementation Guide for the Australian Government.*

Review and Audit

Undertake an iOS post implementation review.

Agencies that deploy iOS devices must undertake a post implementation review. This may assist in identifying policy and implementation inconsistencies and assess the mitigation controls for completeness against the Security Risk Management Plan (SRMP), the System Security Plan (SSP), Standard Operating Procedures (SOP) and the implementation of email protective marking controls. This review must be completed within twelve months of the live production deployment.

Audit compliance with policies and standards for the use of iOS devices.

Setting out policy without monitoring compliance is unsound practice. There should be appropriate internal and – from time to time – external checks of compliance with policies regarding the use of mobile devices. There should also be regular reviews of internal policies, to test their currency and adequacy.

CHAPTER 10

Example Scenarios

This chapter describes hypothetical scenarios showing how the various techniques can be combined.

Unclassified kiosk

An art gallery wishes to use iPod touches as an interactive tour guide for Unclassified information at a specific site. The tour guide information is largely contained within a single app.

The gallery purchased an Enterprise Developer Agreement, and uses this to code-sign the app they have had developed by a contractor.

The gallery set up a Wi-Fi network for the site, and uses a provisioning computer with Apple Configurator to supervise and then “lock to” their developed app. Devices can be deployed, managed and reset with minimal effort.

Unclassified (DLM) BYOD

An agency has decided to allow limited corporate network access to employee owned devices. Users are required to enrol their device in to the agency MDM and agree with an acceptable use policy.

The agency uses a 3rd party MDM server to enforce Configuration Profile restrictions and audit devices for compliance. Non-compliant devices have their MDM profile and associated managed Apps and data revoked immediately. Configuration Profiles take advantage of the Managed Open-In function to prevent movement of documents from agency managed Apps and email to user personal Apps and email.

A Wi-Fi network configured according to relevant ISM government system controls for UNCLASSIFIED with DLM is deployed on premise, permitting limited corporate network access via VPN and limited personal use through an authenticated proxy. Email is provided using the native iOS Mail app using Exchange Active Sync (EAS) over TLS.

PROTECTED with limited personal use

An agency has decided to issue iOS devices for work and limited personal use. The users require access to PROTECTED email and attachments as well as access to their PROTECTED intranet. The agency permits user installation of App Store Apps subject to agreement with an acceptable use policy. Personal email is permitted using the iOS native mail app.

In this case, the agency uses an MDM server, a VPN concentrator for remote access, Exchange for email, a third party gateway filter and Apple Configurator to place devices in to supervised mode. The iOS devices use a client certificate for authentication to Exchange, and a client certificate is used for On-Demand VPN authentication. Configuration Profiles take advantage of the Managed Open-In function to prevent movement of documents from agency managed Apps and email to user personal Apps and email. The third party gateway filter is configured to implement protective markings on email sent from devices. A Wi-Fi network configured according to relevant ISM government system controls for UNCLASSIFIED with DLM is deployed on premise, permitting corporate network access via VPN and limited personal use through an authenticated proxy.

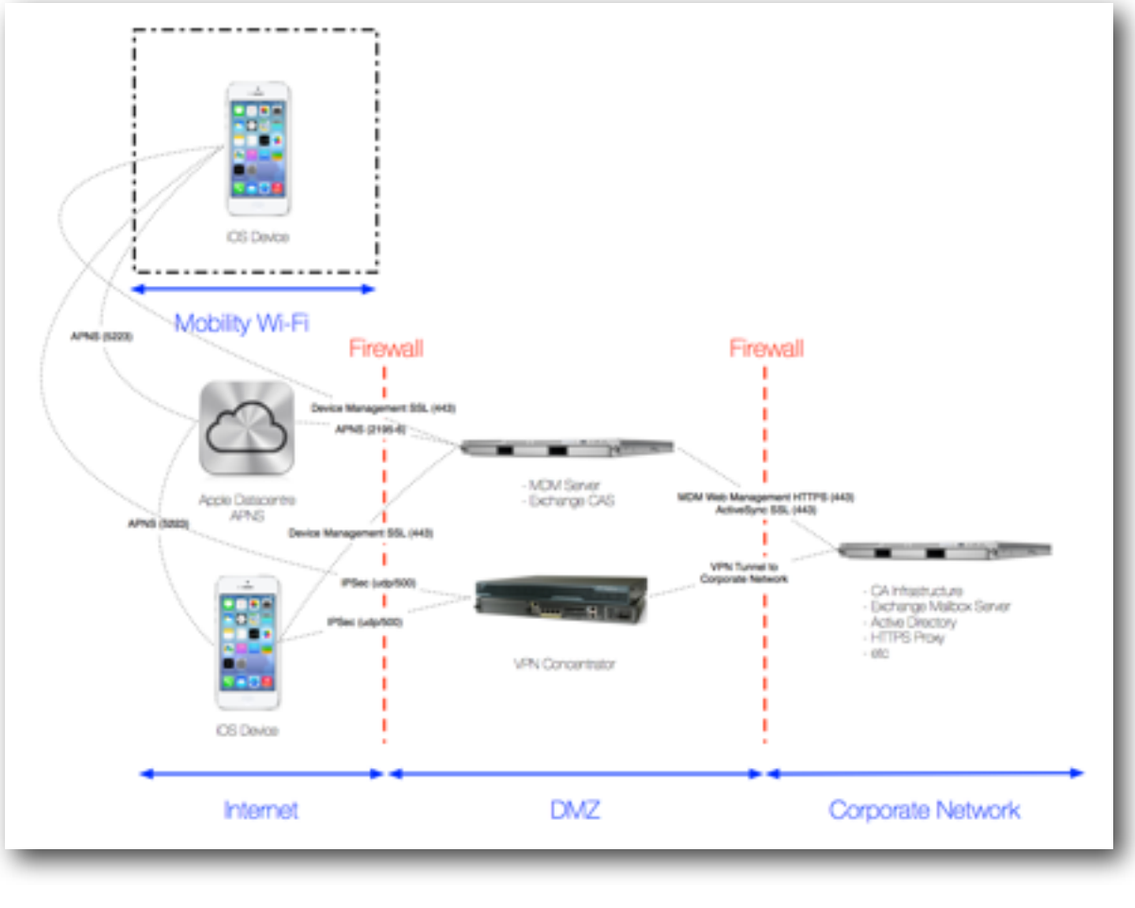


Figure 10.1: Example network diagram for limited personal use

PROTECTED business only

An agency has decided to issue iOS devices for its mobile fleet. Its users require access to their PROTECTED email and attachments, as well as access to a PROTECTED intranet.

The IT team will use Apple Configurator to configure the devices before they are issued to users. The devices will be configured as supervised devices and will be pre-enrolled with the agency's MDM server.

The devices connect to the agency's exchange server using a client certificate for authentication. The IKEv2 VPN is configured as "Always On" with certificate authentication; this forces all traffic over the VPN. Access to internal web resources on the corporate intranet is allowed through an authenticated proxy.

The agency requires all users to sign an acceptable use policy that requires them to install OTA updates when they are available. Compliance will be monitored by the IT team using the MDM.

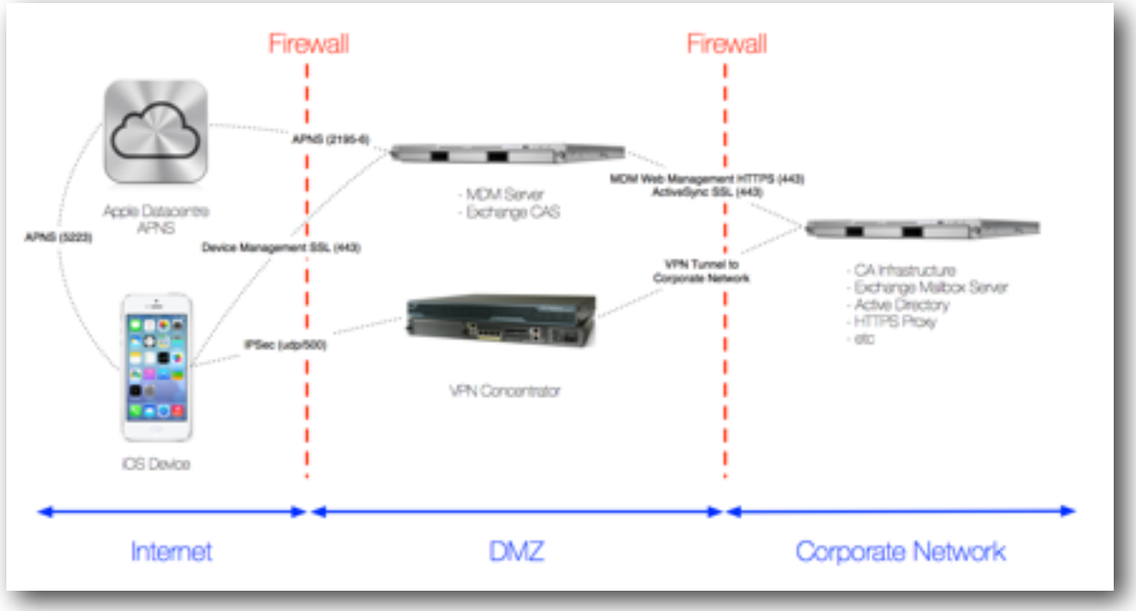
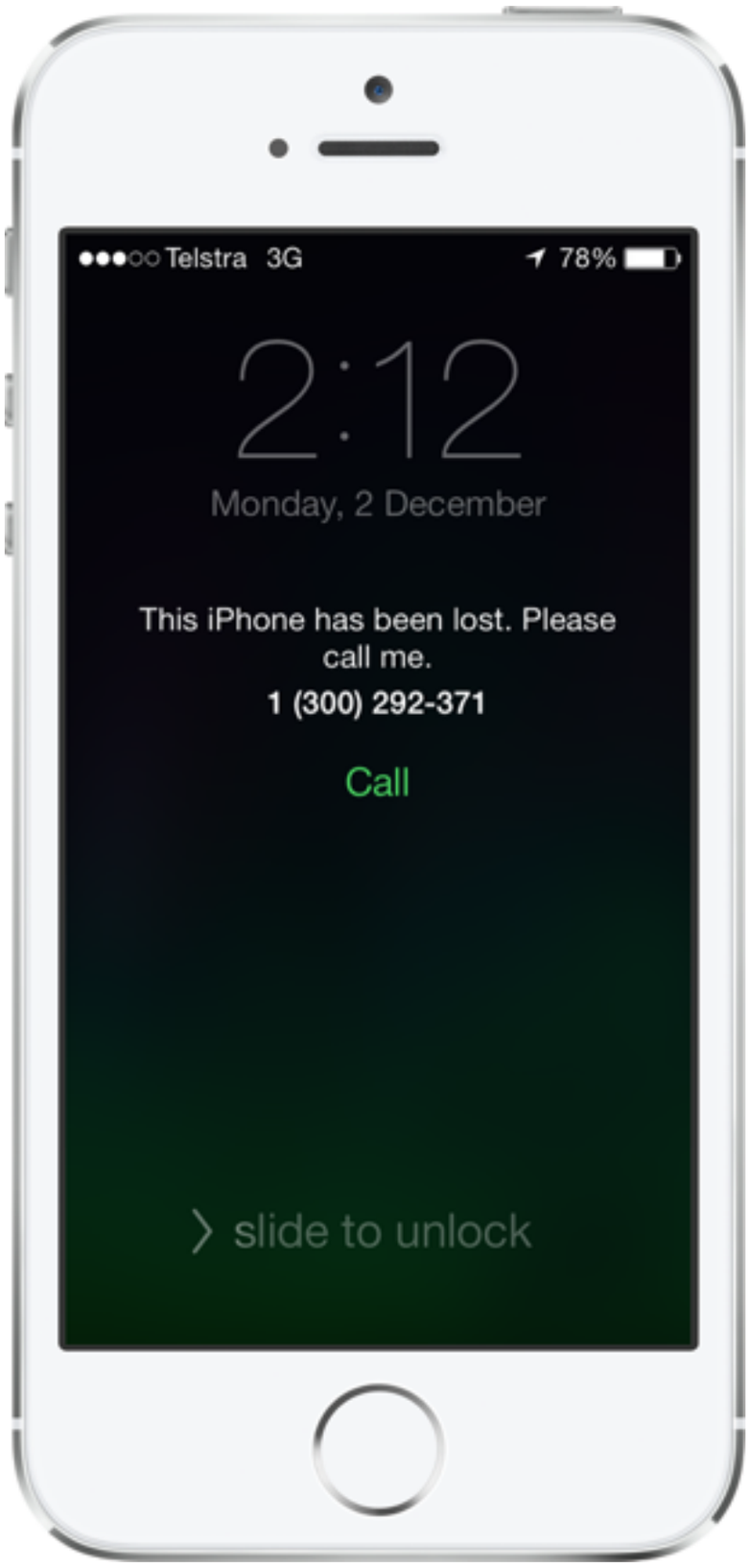


Figure 10.2: Example network diagram for limited personal use

CHAPTER 11

Risk Management Guide



This chapter provides a guide to typical risks associated with mobile devices and recommended mitigation measures.

Australian Government Information Security Manual (ISM)

This chapter should be read in conjunction with the ISM, available from the ASD website:

http://www.asd.gov.au/infosec/ism/

Currently, not all ISM requirements can be implemented on iOS 9 devices. Risk mitigation measures are provided in this chapter for such cases.

Mobile Device Risks

Typical risks, the recommended mitigation measures and the pre-conditions for those mitigation measures are covered in the table below. There are several residual risks in ISM policy that cannot be completely mitigated by technical controls. Agencies will need to assess, accept and manage any residual risks and develop appropriate policy guidance.

iOS does not have a local firewall. The risks associated with this are significantly mitigated by the sandboxed runtime environment in iOS, where persistent background process execution is extremely limited, so risk exposure primarily occurs while an App is active. Use of IPSec IKEv2 VPN, either in an Always-on or per-App configuration can be used to prevent an attacker on the local network targeting the device or a specific app respectively. Use of Always-on VPN is the strongest posture to take in isolating the device from local network attacks.

iOS allows the user to deliberately connect to an untrusted Wi-Fi network. Note that iOS devices will not auto-connect to any unknown Wi-Fi network. The use of IPSec IKEv2 Always-on or per-App VPN configuration is a significant mitigation to the associated risks, as both fail closed. Use of Always-on VPN is the strongest posture to take in isolating the device from local network attacks.

iOS allows the user to deliberately enable or disable the radio transceivers (e.g. Wi-Fi, Bluetooth) in the device. Whilst there are inherent mitigations in the Bluetooth stack and runtime environment, there is no method for a Configuration Profile to force a radio transceiver off. MDM can send a command to disable some radios, but the user can manually enable these again. The only mitigations available at this time are user education, AUP or hardware modification (the latter being permanent, occasionally destroying the device in the process, and voids warranty.).

Risk	Mitigations	Implied Preconditions
Device lost, still on network	Supervised mode with pairing blocked, Strong passcode, data protection enabled, remote wipe, MDM w/ Lost Mode, Find My iPhone/iPad.	Configuration Profiles, EAS or MDM Server in a network reachable location or iCloud account.
Device lost, off network	Supervised mode with pairing blocked, Strong passcode, local wipe, data protection enabled.	Passcode requirement via config profile
Device lost, casual access attempt	Strong passcode, local wipe, data protection enabled.	Passcode requirement via config profile, supervised to prevent pairing (iOS 9 devices do not pair with a new host when locked)
Device lost, forensic access attempt without passcode knowledge	Strong passcode, local wipe, use of supervised mode with pairing blocked, data protection enabled, app usage of appropriate data protection class.	Configuration Profiles, device running up to date iOS. Device in supervised mode with pairing restricted. Use of Devices with A7 or newer processor.
Jailbreaking	Strong passcode, data protection enabled, use of devices with A5 or later processor, use of MDM with DEP, use of supervised mode with pairing blocked, Restriction to prevent untrusted Enterprise Apps being installed, AUP should prohibit jailbreaking.	MDM or Apple Configurator. Use of devices with A7 or newer processor. Keeping iOS versions on devices up-to-date
Malicious runtime code	Code signing, memory and filesystem sandboxing, no-execute heap, disable user-added applications, do not jailbreak operational devices and (for App Store Apps) the App Store review process.	In-house application development capability, CA infrastructure. May mitigate on lower security levels by “approved” lists and MDM monitoring as mitigation.

Table 11.1: Risk management guide

Risk	Mitigation	Implied Preconditions
Users cut and paste agency data into a public email account (e.g. Yahoo or Gmail) and send it from the device.	Disable the creation of separate email accounts, and restrict access to webmail via custom APN and agency proxy, disable screen shots on device via Configuration Profile, filter sensitive mail or attachments at the EAS gateway, use of VDI for sensitive email, contain agency email to a third party email app container.	Configuration Profiles, use of agency proxy. Note that any data that is displayed on the screen of any device can be photographed or video recorded by a camera, and sent via other means. This kind of leakage by deliberate action generally cannot be mitigated against for a mobile device.
Users cut and paste sensitive data from a managed app to an unmanaged app.	Accidental cut-and paste partially mitigated by iOS UI. Ensure that enterprise Apps take advantage of named pasteboards, or disable use of copy paste. Explain risk of accidental disclosure of classified information via copy/paste in AUP.	Custom Apps for named pasteboards. For App Store Apps, agencies can engage with developers directly to procure custom builds under the B2B store
Untrusted devices connect to agency network.	Use of 802.1X NAC, IPSEC or TLS VPN, encrypted VDI.	Use of 802.1X with CA & NAC on Wireless, VPN on Demand with client certificates for agency network access, use of TLS reverse proxy for low security data.

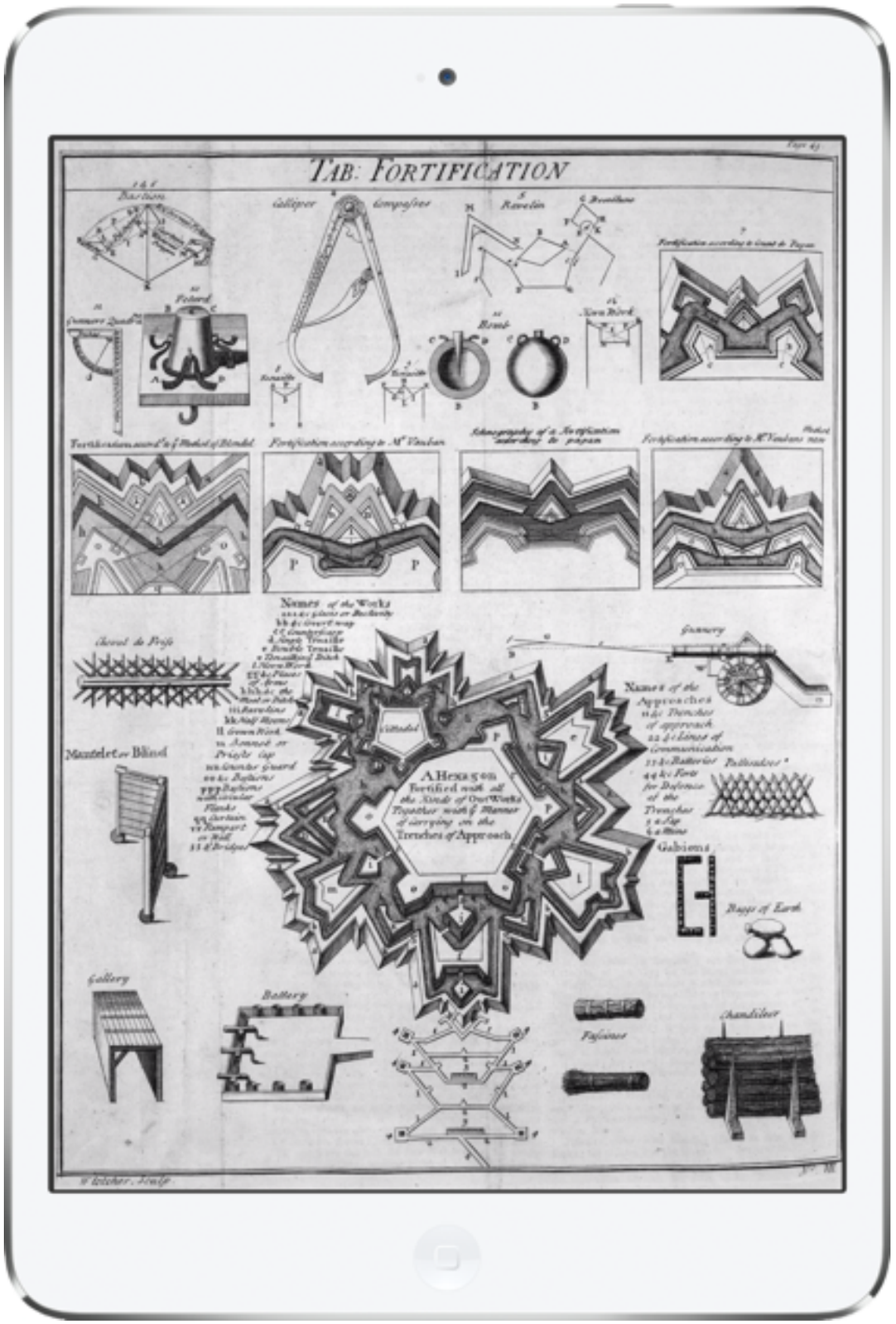
Table 11.1 (continued): Risk management guide

Risk	Mitigation	Implied Preconditions
Data compromise via host computer backup	Force encrypted backup profile restriction, user education, physical security of backup host. Use supervised mode and prevent host pairing	TLS CA infrastructure to sign and encrypt profiles into agency chain of trust.
Data compromise via Bluetooth	iOS only includes 5 or 7 of the 35 Bluetooth profiles, depending on device. This strictly limits functionality. For included profiles see: http://support.apple.com/kb/HT3647	Apps that share information outside the system supplied profiles via Bluetooth (e.g using Multi-peer gaming APIs) should be individually evaluated.
Accidental disclosure of classified information via iMessage	Have users create agency specific iCloud account, rather than personal account. Educate users of the risk of accidental disclosure of classified information via “Open-In” iMessage in AUP. Disable iMessage via Configuration Profile restriction. Provide alternative chat environment.	Agency needs to decide if the risk of accidental disclosure of classified information, mitigated by Open-in restrictions is greater than the utility of iMessage.
SMS message interception by hostile telecommunications infrastructure	Allow iMessage use via Configuration Profile.	Contact ASD to discuss particular user travel circumstances.

Table 11.1 (continued): Risk management guide

CHAPTER 12

Firewall Rules



Allow required iOS functionality while preserving the security of your network.

Firewall Rules

Several firewall rules may need to be implemented to allow correct functionality. Depending on what functionality is required from iOS devices, MDM servers and iTunes, several firewall rules may need to be implemented.

Firewall Ports

iTunes and iOS devices may need firewall rules adjusted, depending on the functionality required, or allowed, on an intranet. Typically, devices need outbound access to Apples's network (17/8), and Verisign's OCSP URL. Note that most traffic to Apple servers from iOS itself is certificate pinned to Apple. The main knowledge base articles describing ports required by Apple devices are given below, with a summary around iOS and iTunes in Table 12.1 (below):

http://support.apple.com/kb/TS1379

http://support.apple.com/kb/TS1629

Destination host name	Destination IP	Port	Reason
ocsp.apple.com	17.0.0.0/8	443	Online Certificate Status for code signing certificates
crl.apple.com	17.0.0.0/8	443	Certificate Revocation List for codesigning certificates
gateway.push.apple.com	17.0.0.0/8	2195	Apple Push Notification Service
feedback.push.apple.com	17.0.0.0/8	2196	Apple Push Notification Service
phobos.apple.com	17.0.0.0/8	80, 443	iTunes Store, Device activation
itunes.apple.com	17.0.0.0/8	80, 443	iTunes Store, Device activation
deimos.apple.com	17.0.0.0/8	80, 443	iTunes U
deimos3.apple.com	17.0.0.0/8	80, 443	iTunes Music Store and album cover media servers.
ax.itunes.apple.com	17.0.0.0/8	80, 443	iTunes Store, Device activation
gs.apple.com	17.0.0.0/8	80, 443	iTunes Store, Device activation
albert.apple.com	17.0.0.0/8	80, 443	iTunes Store, Device activation
ax.init.itunes.apple.com	17.0.0.0/8	80, 443	Device activation
evintl-ocsp.verisign.com	199.7.55.72	80, 443	Digital signature verification for iTunes content
evsecure-ocsp.verisign.com	199.7.55.72	80, 443	Digital signature verification for iTunes content
a1535.phobos.apple.com	17.0.0.0/8	80, 443	iTunes Music Store and album cover media servers.

Table 12.1: Firewall rules.